

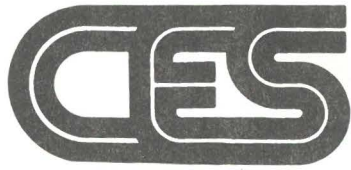


Ed-Lab 700
EXPERIMENT MANUAL

DIGITAL SYSTEMS

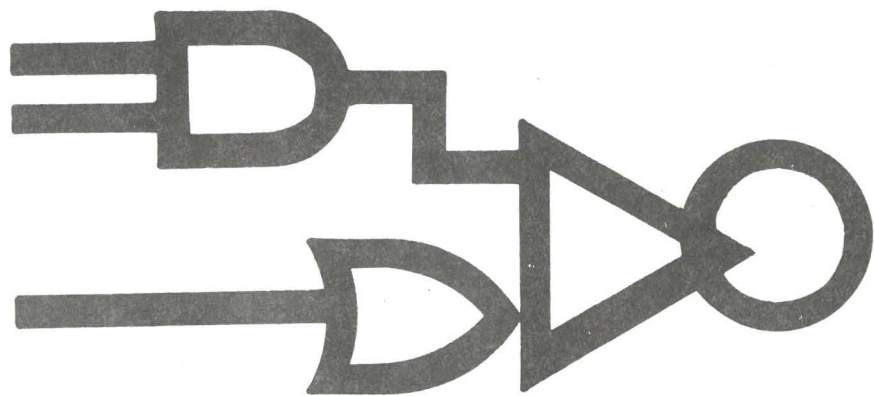


CES INDUSTRIES, INC.



Ed-Lab
EXPERIMENT MANUAL

DIGITAL SYSTEMS



CES INDUSTRIES, INC.

This manual is the property of:

Name	_____
Class	_____
School	_____

EXPERIMENT MANUAL

DIGITAL SYSTEMS

Copyright © 1971, 1972, 1973, 1975, 1979 CES INDUSTRIES, INC.

All rights reserved. No part of this book may be duplicated or reproduced in any form without the specific written permission of CES Industries, Inc. The use of this manual and the experiments described is restricted to performance with CES Ed-Lab training equipment and is prohibited from being used with any other training aid or electronics components or for other educational purposes. All rights are reserved for use on television. CES makes no representation that the circuits, logic, or programs described herein will not infringe on existing or future patent rights. There is furthermore no implication in the granting of licenses to make, sell, or use any circuits, logic, or programs described herein.

International Standard Book Number 0-86711-001-5
Printed in the United States of America

CES INDUSTRIES, INC.
130 Central Avenue
Farmingdale, New York 11735

PREFACE

The advances in the electronic, computer, and electromechanical technologies during the past decade have made possible a modern computer industrial revolution capable of transporting men to the moon and back. The use of computer electronics in business and the government here on earth has resulted in increased production and efficiency in many industrial processes, and in many related consumer areas including the communications, entertainment, medical, education, construction, clothes, and food industries. The result has also been to create many new jobs in the design, manufacturing and servicing of computers and automation equipment. Our challenge is both to improve the effectiveness of the computer technology as well as the application of this technology as a tool for the further improvement of the life and well being of all.

The aim in developing the Ed-Lab training equipment has been to provide a unit which covers the broadest range of computer subjects including logic, digital systems, conversion between analog and digital signals, data processing, control amplifier applications, and computer circuitry. The result is the Ed-Lab System which holds the interest of the student and teacher by providing an almost limitless potential for learning and creating. We at CES use this equipment in the actual design and test of new digital and computer equipment and find the learning a continuing enjoyment.

A second aim was to provide a rugged and professional trainer capable of continuous and reliable use in the classroom. Instrument leads and jacks and sturdy panels and construction are used and there are no loose parts to get damaged or lost. Use of this Experiment Manual does not require any previous experience by either the student or the instructor. The material is separated into the following sub-sections: BASIC DIGITAL CONCEPTS, LOGIC TECHNIQUES, ARITHMETIC TECHNIQUES, SEQUENTIAL SYSTEMS, BASIC ANALOG CONCEPTS, CONVERSION TECHNIQUES, and COMPUTER CIRCUITS. Experiments are self-contained and students are urged to tackle any experiments and to proceed at their own rate. We know that you will devise many new and interesting experiments and applications.

This manual has been written to convey the technical material that we have found to be both most useful in the understanding and design of digital and computer equipment, and most compatible with the educational curricula and goals. The experiments are written in a practical fashion with all the necessary descriptive material and exercises included. Experiments and exercises are textbook compatible. The Computer Systems Lab. - Ed Lab 700 and this manual will provide you with the knowledge and experience which we hope will be invaluable in your future endeavors.

Norman Nesenoff
President
CES INDUSTRIES, INC.
Educational Division

Safety is good common sense. The Ed-Lab equipment is designed with all current limited and voltage limited Power Supplies so that there is no danger to the student when touching any wires and no danger of burning-out any of the electronic components.

DO NOT USE AN EXTERNAL POWER SUPPLY WITH THE ED-LAB.

While working in the laboratory, the student is exposed to many different types of equipment, power tools, soldering irons, test equipment, drills, rotating machinery, and so on. The 120 volt power socket can cause a shock, and the shock can result in your jumping or falling. Care should be exercised while working on any equipment including automobiles, television sets, carpentry tools, as well as electricity and electronics.

Do not get involved in any "horsing-around" or in practical jokes. Don't throw anything in the lab. Don't drop the equipment off the bench or on your foot. Don't burn the line cord with a soldering iron. If you remove the Ed-Lab front panel don't touch the 120 volt line-cord connections in the rear while the plug is still connected to the socket.

If an accident occurs, notify your instructor, shut off all power, and administer first aid as necessary. Your instructor should review with you the safety requirements in your laboratory.

If you observe an accident involving electrical shock,
DON'T JUST STAND THERE - DO SOMETHING!

RESCUE OF SHOCK VICTIM

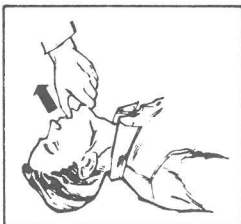
The victim of electrical shock is dependent upon you to give him prompt first aid. Observe these precautions:

1. Shut off the high voltage.
2. If the high voltage cannot be turned off without delay, free the victim from the live conductor. REMEMBER:
 - a. Protect yourself with dry insulating material.
 - b. Use a dry board, your belt, dry clothing, or other non-conducting material to free the victim. When possible PUSH - DO NOT PULL the victim free of the high voltage source.
 - c. DO NOT touch the victim with your bare hands until the high voltage circuit is broken.

FIRST AID

The two most likely results of electrical shock are: bodily injury from falling, and cessation of breathing. While doctors and pulmotors are being sent for, DO THESE THINGS:

1. Control bleeding by use of pressure or a tourniquet.
2. Begin IMMEDIATELY to use artificial respiration if the victim is not breathing or is breathing poorly:
 - a. Turn the victim on his back.
 - b. Clean the mouth, nose, and throat. (If they appear clean, start artificial respiration immediately. If foreign matter is present, wipe it away quickly with a cloth or your fingers).
 - c. Place the victim's head in the "sword-swallowing" position. (Place the head as far back as possible so that the front of the neck is stretched).
 - d. Hold the lower jaw up. (Insert your thumb between the victim's teeth at the midline - pull the lower jaw forcefully outward so that the lower teeth are further forward than the upper teeth. Hold the jaw in this position as long as the victim is unconscious).
 - e. Close the victim's nose. (Compress the nose between your thumb and forefinger).
 - f. Blow air into the victim's lungs. (Take a deep breath and cover the victim's open mouth with your open mouth, making the contact air-tight. Blow until the chest rises. If the chest does not rise when you blow, improve the position of the victim's air passageway, and blow more forcefully. Blow forcefully into adults, and gently into children.
 - g. Let air out of the victim's lungs. (After the chest rises, quickly separate lip contact with the victim allowing him to exhale).
 - h. Repeat steps f. and g. at the rate of 12 to 20 times per minute. Continue rhythmically without interruption until the victim starts breathing or is pronounced dead. (A smooth rhythm is desirable, but split-second timing is not essential).



DON'T JUST STAND THERE - DO SOMETHING!

CONTENTS

UNIT NO. 1

BASIC DIGITAL CONCEPTS

1. Introduction
 - Power Inputs
 - Input Switches
 - Display Lamps
 - Tables
 - Trainer Summary
 - One Zero
2. Binary Notations
 - Coding
3. Logical NOT
 - Logic Symbol
 - Logic Problem
4. Logical AND
 - Truth Table
5. Logical OR
6. Logic Problem
 - Logic Diagram
7. Combination Logic
8. Logic Expressions
9. Majority Logic System
 - Minority Logic
 - Voting System
10. Comparison Logic
 - Inequality Logic
11. Parallel Output Logic
 - Collector Logic - DTL
12. Exclusive - OR
 - Inclusive - OR
13. Equality Comparison Logic
14. Logic Design
15. Logic MAP
16. VENN Diagram
17. Logic Memory
 - Logic Feedback
18. Sequential Logic
 - Set - Clear Memory
19. Logic Diagrams and Equations
 - Logic Equation Operators
 - NAND
 - NOR
20. Computer System

LOGIC TECHNIQUES

21. NAND Logic Element
 - Universal Logic
 - NAND : AND - OR - NOT
22. NAND Applications
 - Parallel Outputs
 - Exclusive - OR
 - Expander
23. NAND Systems
 - AND - OR
 - Logic System
 - NAND Memory
 - Series NAND
24. NOR Logic Element
 - Universal Logic
 - NOR : AND - OR - NOT
25. NOR Applications
 - Expander
 - Parallel Outputs
 - NOR Memory
26. NOR Systems
 - Series NOR
 - OR - AND
 - AND - OR
27. Combined Logic System
28. Boolean Algebra
 - Logic Rules and Laws
 - Tautology
29. Negative Logic and Duality
 - DeMorgan's Theorem

CONTENTS
UNIT NO. 2

ARITHMETIC TECHNIQUES

- 30. Binary Numerals
 - Binary Coding
 - Octal Coding
- 31. Code Conversions
 - Binary to Numerical Converter
 - Binary to Decimal Conversion
- 32. Parity Codes
 - ODD Parity Decoder
 - EVEN Parity Encoder
- 33. Digital Codes
 - Binary Coded Decimal
 - Excess Three
 - Weighted Codes
 - Five Bit Codes
 - BCD to 2421 Converter
- 34. Gray Code
 - Gray to Binary
 - Binary to Gray
- 35. Binary Addition
 - Half Adder
 - NAND Half Adder
 - NOR Half Adder
 - Subtraction
- 36. Serial Full Adder
- 37. Parallel Adder
- 38. Negative Numbers
 - One's Complement
 - Two's Complement
 - Subtraction
 - Fractions
 - Powers of Two
- 39. Binary Multiplication
- 40. Other Number Systems
 - Radix
 - Ternary Memory

SEQUENTIAL SYSTEMS

- 41. Flip - Flops
 - Set - Clear R - S
 - Symbol
 - Type D
 - J - K - Trigger
 - Sequential Operation
- 42. Storage Register
 - Data Transfer
- 43. Clock and Triggers
 - Triggered Flip - Flop
 - Clock Rate
- 44. Binary Counters
 - UP Counter
 - DOWN Counter
- 45. Counter Decoders
- 46. Counter Presets
- 47. Manual Counting
- 48. Interval Counter
 - Time Duration
- 49. Rate on Frequency Counter
- 50. Steered Inputs/Shift Register
- 51. Register Counters
 - Johnson Counter
 - Ring Counter
 - Four Phase Clock
 - Two Phase Clock
- 52. BCD Counter
 - Duodecimal Counter
- 53. Modulo Counters
 - Modulo 5/3/6
- 54. Decode Strobing
 - Timing
- 55. Adder With Memory
 - Gray to Binary Converter
 - Binary to Gray Converter
- 56. Counting Systems
 - UP - DOWN Counter
 - Synchronous Counter
- 57. Arithmetic Unit
 - Serial Accumulator
 - Data Entry
 - Addition
 - Subtraction

CONTENTS

UNIT NO. 3

BASIC ANALOG CONCEPTS

- 58. Analog Data
 - Meter
 - Voltages
 - Scale Factor
 - Resolution
 - Decibels
 - Data Graphs
 - Binary Levels
 - Tolerances
 - Potentiometers
- 59. Analog Comparator
 - Operational Amplifier
- 60. Threshold Logic
- 61. Schmitt Trigger
 - Level Detection
 - Hysteresis
 - Variable Threshold
- 62. Analog Amplification
 - Gain
 - Linearity
 - Transfer Function
 - Operating Range
- 63. Operational Amplifiers
 - Feedback
 - Scaled Inverter
 - Follower
 - Attenuation
 - Positive gain
 - Negative gain
 - Integration
 - Linear Rectifier
 - Multiplication
 - Division
- 64. Difference Amplifier
 - Gain
 - Common Mode
 - Common Mode Rejection Ratio
- 65. One Shot
- 66. Solid-State-LSI Logic
 - NOT, NAD, NOR,
 - AND, OR, FLIP-FLOP
- 67. Capacitor-Diode Gate

CONVERSION TECHNIQUES

- 68. Digital to Analog Conversion
 - Scale Factor
 - Offset
 - Non-Linearity
 - Resolution
- 69. Analog to Digital Conversion
 - Counting
 - Half-Split
 - Feedback
- 70. Digital Voltage Sweep
 - Period
 - Frequency
- 71. Analog Gates & Multiplexer
 - Sequencing
 - Sampling rate
- 72. Follower and Hold
 - Gain
 - Resistance
 - Loading
 - Drift Rate
- 73. Digital Voltmeter
 - Scale Factor
 - Calibration
- 74. Data Acquisition System
 - Multiplexer
 - A/D Conversion
 - Sequence Counter
 - Control Timing

CONTENTS

COMPUTER CIRCUITS

- 75. Integrated Circuit Logic
 - Diode Logic
 - DTL Diode Transistor Logic
 - TTL Transistor - Transistor Logic
 - RTL Resistor - Transistor Logic
 - ECL Emitter Coupled Logic
 - HTL High Threshold Logic
 - Dual In-Line Integrated Circuits
- 76. Solid - State Electronics
 - Meter
 - Ohm's Law
 - Kirchhoff's Voltage Law
 - Kirchhoff's Current Law
 - Potentiometric Voltmeter
 - Series Resistors
 - Parallel Resistors
 - Voltage Divider
 - Flip-Flop
 - Monostable Multivibrator
 - Exclusive - NOR
 - Voltage Clamp
 - Diode Characteristics

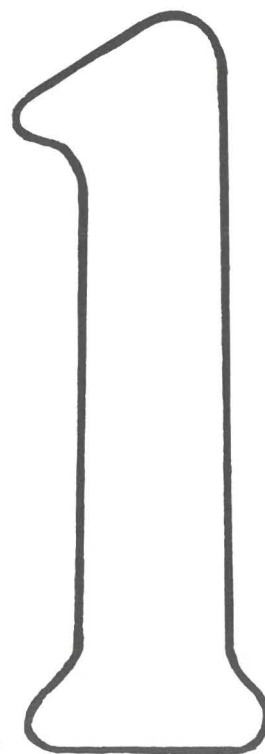


Ed-Lab

EXPERIMENT MANUAL

BASIC DIGITAL CONCEPTS and LOGIC TECHNIQUES

UNIT NO.



CES INDUSTRIES, INC.

EXPERIMENT MANUAL

Copyright © 1974, 1976, 1978, 1980, 1981 CES INDUSTRIES, INC.

Name _____

Course _____

OBJECTIVE:

This experiment introduces the operation of the Ed-Lab 700 Digital Trainer and the use of this Experiment Manual and the operation of the Switches, Lamps, and Patch Cords.

DESCRIPTION:

The Ed-Lab trainer includes the "training panel" and "connection patch-cords". Any incorrect wiring will not damage any parts, so do not hesitate to use the equipment and make connections.

PROCEDURE:

Connect the "line-cord" to the electrical socket to provide electrical power to the Ed-Lab Trainer. The small Pilot Light should be ON. (Earlier models do not have an on-off Switch or a Pilot Light.)

This first experiment consists of connecting a SWITCH to a LAMP using a PATCH CORD or WIRE. A sketch of the wiring is shown below in Figure 1.

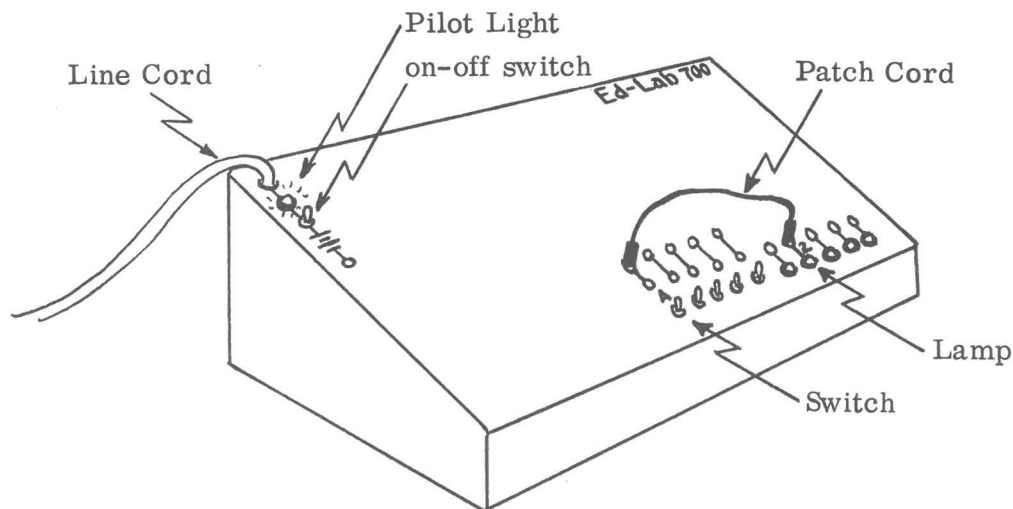
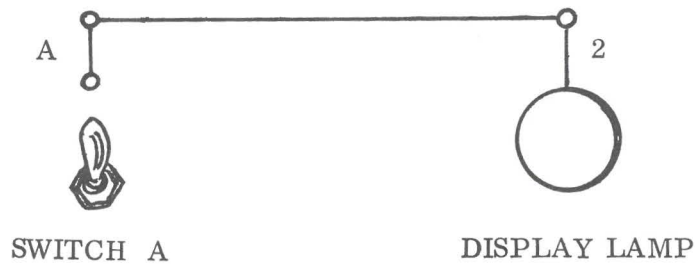


Figure 1. SWITCH - LAMP WIRING

On the lower right hand side of the panel are 5 switches and 5 lamps. A wire is shown above connected between the Switch marked "A" and the Lamp marked "2". A simpler sketch or SCHEMATIC DIAGRAM is shown on the following page in Figure 2.



Experiment Figure 2. Switch - Lamp Schematic Diagram in Figure 2.

Connect the Wire as shown in the above diagram.

When the input SWITCH is UP, there will be a positive voltage on the wire and the output LAMP will be ON. When the SWITCH is DOWN, there is zero voltage on the wire and the LAMP is OFF. Operate the Switch UP and DOWN.

Does the circuit operate in accordance with the description? _____.

We use the Switch as an INPUT, and the equipment operates the Lamp as an OUTPUT.

When conducting any experiment, it is necessary to enter the results of the test in a chart or Table as shown below.

Input	Output
SWITCH A (UP OR DOWN)	LAMP 2 (ON OR OFF)
UP	
DOWN	
UP	
DOWN	

Operate the Switch and enter the output results in the table (ON or OFF).

Repeat this Experiment using the table below. A "short-hand" notation is used. The column marked "A" refers to Switch A. The column marked L2 refers to LAMP 2.

The symbol "1" (i. e. ONE) refers to UP for the Switch and to ON for the Lamp.

The symbol "0" (i. e. ZERO) refers to DOWN for the Switch and OFF for the Lamp. Enter the data below:

SHORT HAND TABLE

A	L2
1	
0	
1	
0	

This first experiment has been presented as the simplest experiment possible, that is, using one switch, one wire and one lamp. The connections were shown in a picture form (Figure 1) and in a schematic diagram (Figure 2). The rest of the experiments will only be shown with the schematic diagrams. Digital systems require many wires, however, it is no more difficult than adding one wire at a time.

Each experiment requires reading a Description and Procedure, Wiring the Experiment, Operating the Inputs, Reading the Outputs and Entering the results in a Table. Questions are then presented. The students are to answer the questions and submit a report with the Table of experiment results, and the Answers to the questions.

The Question given on the following page requires a 100% increase in complexity; i. e., it uses 2 wires.

QUESTIONS:

1. A diagram is given below for two input switches and two output lamps.

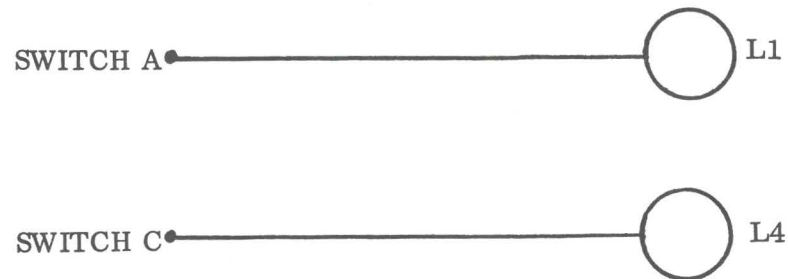


Figure 2 TWO SWITCH SCHEMATIC

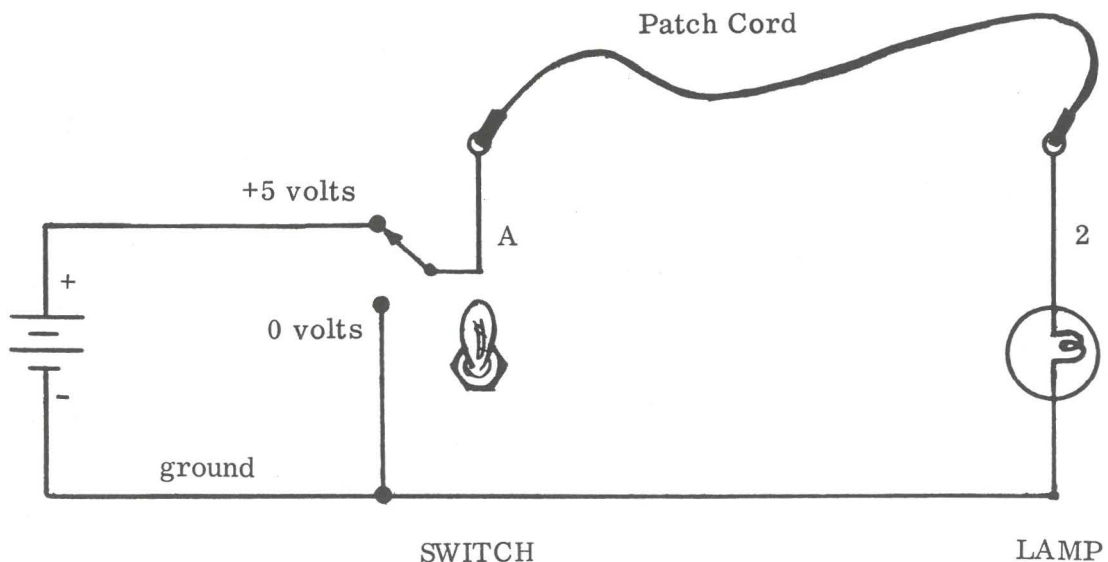
Fill in the Table below with expected outputs for all possible combinations of inputs.

INPUT		OUTPUT	
A	C	L1	L4
0	0		
0	1		
1	0		
1	1		

Electrical Data For Review: IT IS NOT NECESSARY TO READ THIS PAGE.

The following describes the switch and lamp electrical circuit.

Five switches marked A, B, C, D and E are located on the lower center of the panel. Above each switch are two black jacks connected by a white line. These are the switch outputs. When the switch is in the up position, 5 volts D.C. is connected to the output pins. When the switch is down, a ground(zero volts) is connected to the output pins. The switch is internally wired to perform these connections as shown below.



In digital logic, only two signals are important, 5 volts and ground, so these signals are permanently wired to the switches and only the output is available.

The lamps marked 1, 2, 3, 4 and 5 are located on the lower right side of the panel. Each lamp is controlled by the jack above it. When a voltage is applied to the lamp jack, the lamp will light and be ON. The lamp will be out when the input jack is at ground. (The lamps have driver transistor circuits built in so that they do not need much input power to be lighted.)

TRAINER SUMMARY: IT IS NOT NECESSARY TO READ THIS SUMMARYNOTE

The summary of all elements on the Ed-Lab Trainer is given in the following 4 pages for reference. It is not necessary to read this summary. Each element is described later in separate experiments.

1. Power Supply

Power for the CES 700 panel is obtained from 115V AC. (In earlier models, batteries or an external power supply are used. The AC adapter supplied with the unit permits operation on 115V, 60 Hz. The output of the adapter is plugged into the panel power supply jack. A standard 6 volt battery can be used with an adapter plug. If an external power supply is used with this adapter plug, use 5 volts. Terminal P (right side of meter) provides a power supply voltage between +5 and +6 volts through a current limiting circuit. This output is used to provide power to ADD-ON units.

The power input to the panel is internally connected to power all circuits. The "1" logic level voltage is zener regulated to approximately 3.8 volts. The "0" logic level is ground. See Experiment No. 1.

2. Meter

The meter has a 50 microampere movement with a resistance of approximately 1000 ohms. The - and +I jacks are directly connected to the meter. The short interconnecting lead should be connected between these two jacks when the panel is not in use to protect the meter movement. These jacks (- and +I) are used for current measurement (50 microamps) and for very small voltage measurement in analog applications. This provides a 0.05 volts scale corresponding to the 1000 ohms internal impedance. For 5 volts range operation, connect - jack to GND and connect +E to the terminal to be measured. This scale presents a 100,000 ohm load or 20,000 ohms per volt. See Experiment Nos. 58 and 76.

3. Lamps

The five display lamps have driver inputs and can be used to observe the logic level at the input or output of any of the logic gates and flip-flops.

4. Switches

The five toggle switches are connected to provide a logic 0 (GND) at the output jacks when in the DOWN position and logic 1 (+5V) in the UP position. The push button switch (M) is not connected internally and is used to momentarily switch from one voltage level to another.

5. Potentiometers

The X and Y potentiometers are 1000 ohms each and are used to provide analog voltage levels, or variable resistances.

6. Inverters

The five inverter or NOT circuits provide a "1" output when a "0" input is applied and a "0" output when a "1" input is applied. See Experiment No. 3. All logic is DTL.

7. AND Gates

The four AND gates provide a "1" output when all of the used inputs are connected to a "1" level. Unused inputs may be left open or connected to a "1". See Experiment No. 4.

8. NAND Gates

The operation of the NAND gates is the same as an AND gate with an inverted output. When all used inputs are "1's", the output is "0". All unused inputs may be left open or connected to "1". See Experiment No. 21.

9. OR Gates

The output of an OR gate is a "1" when any one or more of the inputs is "1". All unused OR inputs must be connected to "0" or GND. See Experiment No. 5.

10. NOR Gates

The output of a NOR gate is a "0" when any one or more of the inputs is "1". All unused NOR inputs must be connected to "0" or GND. See Experiment No. 24.

11. EXCLUSIVE -OR

The output of an EXCLUSIVE-OR gate is an "1" when one (and only one) input is a "1". These gates can be used as a NOT gate by connecting the other input to a "1". See Experiment No. 12.

12. Diodes

The three diodes are used to construct diode and diode-transistor circuits and as a voltage clamp.

13. Transistors - SOLID STATE

Three transistors are used to construct basic amplifier and logic circuits and the analog gate for multiplexing. The circuit is an NPN silicon transistor with a 15,000 ohm base resistor. See Experiment No. 75.

14. Resistors

The two resistors, R1 and R2, are 3000 ohms, 5%, 1/4 watt.

15. Threshold Logic

This circuit provides a selection of resistors with a common terminal for use in constructing Threshold logic elements and semiconductor electronic circuits. The resistance values are :

1 to L	15K	5%
2 to L	15K	5%
3 to L	10K	5%
4 to L	5.1K	5%

(Example: From 1 to 2 is 30K, etc.) See Experiment No. 75 and 60.

16. One - Shot

This circuit provides an output pulse of approximately 1/2 second when triggered. When R is connected to a "1", the circuit operates as a lagging-edge one-shot. The normally "0" output goes to a "1" pulse with an input negative going signal. Connecting R through a resistor (Threshold 30K) to "1" increases the pulse width. Connecting R to a "0" provides a leading-edge one-shot. The normal output is "1" and an input changing from 0 to 1 results in a short "0" pulse. See Experiment No. 72.

17. Follower - Hold

This circuit consists of an emitter follower darlington circuit and a 100 microfarad input capacitor. With the capacitor terminal open, the circuit operates normally as an emitter follower. With the capacitor terminal connected to "0", the capacitor will charge to any voltage momentarily applied to the input terminal and the output will "Hold" this level until required. See Experiment No. 72.

18. Schmitt- Trigger

The Schmitt - trigger converts analog levels to binary signals. The output switches level when the voltage increases or decreases to a predetermined level. The level of switching can be altered by connecting the R input to a variable voltage from the arm of a Potentiometer. See Experiment No. 61.

19. Clock

This circuit generates pulses for triggering gates and flip-flops. The speed in the LOW position is one pulse per two seconds; in the HIGH position, it is 1 KHZ. The clock speed can be adjusted on low or high speed positions by connecting the C output through one of the potentiometers to ground. See Experiment No. 43.

20. Flip - Flops

These circuits are R-S and J-K flip flops. For R-S operation, application of a "0" at the $\bar{S}$ input will "set" the Q output to "1". A "0" applied to the $\bar{R}$ input will clear or reset Q to "0". A pulse connected to the T input of the flip-flop will cause the flip-flop to switch from one state to another everytime the trigger input goes from "1" to "0". For J-K operation, the clock may be used as an input trigger. With $J = 1$ and $K = 0$, a clock trigger will "set" the flip-flop to $Q = 1$. With $J = 0$ and $K = 1$, a clock trigger will "clear" the flip-flop to $Q = 0$. With both $J = 1$ and $K = 1$, the output will switch for each input trigger. With both $J = 0$ and $K = 0$, the output will not respond to an input trigger. See Experiment Nos. 41, 43, and 50.

21. Digital - to - Analog Converter

This circuit converts binary inputs (8421 code) to an analog voltage. It is also used in analog-to-digital converter circuits. For good accuracy, it is desirable to use a NOT circuit as a driver at each of the inputs. See Experiment No. 68.

22. Comparator Op - Amp

This circuit is a differential operational amplifier which can be used to study op-amps or as a comparator to provide binary outputs. If the + input is greater than the - input, the output is a "1". If - is greater than +, the output is "0". See Experiment No. 59.

23. Loading Logic gates and flip-flops are DTL integrated circuits. Output load factor is 8 for all except the NOT, which is 7. The input load factor is 1 for all loads except flip-flop trigger "T", which is 2. This means that the output of one gate can be used to drive up to 8 other gate inputs or 4 flip-flop "T" inputs.

OBJECTIVE:

This experiment introduces the idea of CODING the switches and in using BINARY notations in digital logic and in computers.

DESCRIPTION:

The Switch has two positions -- Up and Down; and the Display Lamp has two conditions--On and Off; both using the short-hand notation 1 and 0. The term "bi" means "two" (bicycle - 2 wheels, bi-weekly - every two weeks) and BI-NARY means a system of notation which has two conditions; 1 and 0. (This is derived from the Greek where "bi" means "two" and "nary" means "number".)

The switches and lamps are CODED with "words" such as RAIN. When the switch is UP (binary 1), the condition is YES or TRUE (yes there is rain). When the switch is DOWN (binary 0) the condition is NO, that is (no rain). When the Lamp is ON (binary 1) it also means YES, and when the Lamp is OFF (binary 0) it means NO.

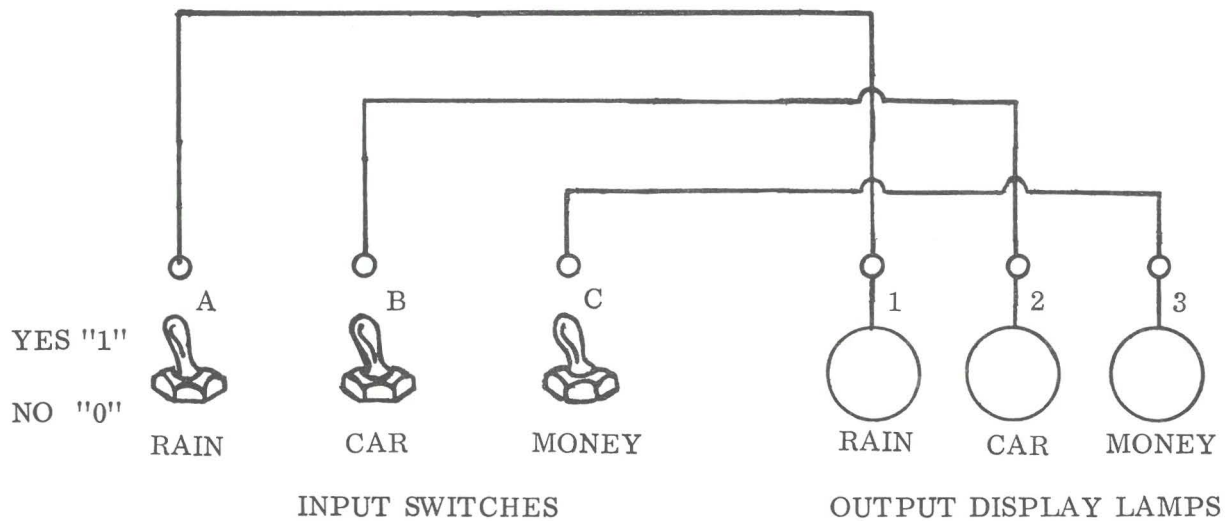
PROCEDURE:

This experiment uses 3 Switches as Inputs connected to 3 Lamps used as outputs.

The Switches are coded as follows:

Switch	Code	
A	RAIN	up means, <u>yes</u> , it is raining
B	CAR	up means, <u>yes</u> , we have a car
C	MONEY	up means, <u>yes</u> , we have money

The switch and lamp diagram for this experiment is shown below:



Experiment No. 2 BINARY CODING

Wire the experiment using the 3 wires as shown above. Set-up the 4 Input conditions given in the Table below. Enter the corresponding lamp outputs in the table.

	INPUTS			OUTPUTS		
	A RAIN	B CAR	C MONEY	1	2	3
1.	1	0	0			
2.	0	0	0			
3.	0	1	1			
4.	1	1	1			
5.						
6.						
7.						
8.						

Three wires or three BINARY DIGITS are used. The first and last pairs of letters are used in a shorthand for BInary digiT^S to form the word BITS. Therefore, this system with three switches, three wires (plus ground), and three lamps can be described as a 3 BIT BINARY CODED DIGITAL COMMUNICATIONS SYSTEM.

Note that the input setting (1) on the previous page is:

RAIN (1), yes, it is raining;

CAR (0), we have no car.

MONEY (0), there is no money.

Do the outputs correspond to the proper inputs? _____

What condition is given by condition 2. above? _____

Fill in all the different possible combinations.
How many are there? _____

QUESTIONS:

1. How many different combinations are there with 3 inputs? _____
2. How many different combinations are there with 2 inputs? _____
3. What would be the input settings if it were not raining, we had a car, and we had no money?

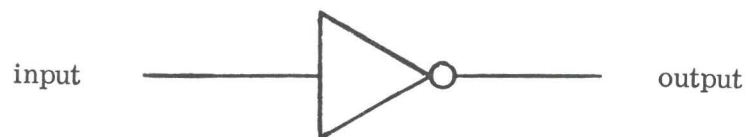
OBJECTIVE:

This experiment demonstrates the operation of the computer logical element known as the NOT element. This element is also called an INVERTER. The LOGIC SYMBOL is also introduced.

DESCRIPTION:

I am not going outside if it is raining. I am not pressing the car accelerator if the light is red. If it is not raining, I am going to the beach. If he is not taking his car, then I am not going.

The Logical NOT operation is essential both in our own logical thinking and in Computer Logic. The symbol for the NOT element is shown below. The triangle indicates that the element is an electronic circuit. The small circle indicates that the function is a NOT. The output is NOT the input. If the input is a 1, the output is a 0. If the input is a 0, the output is a 1.



NOT

LOGIC SYMBOL

Figure 3-1 shows the LOGIC DIAGRAM for this experiment. The diagram has the LOGIC SYMBOL for the NOT circuit, a symbol for the Display and a sketch for the switch. The Logic and Coding for this experiment is for the following statement:

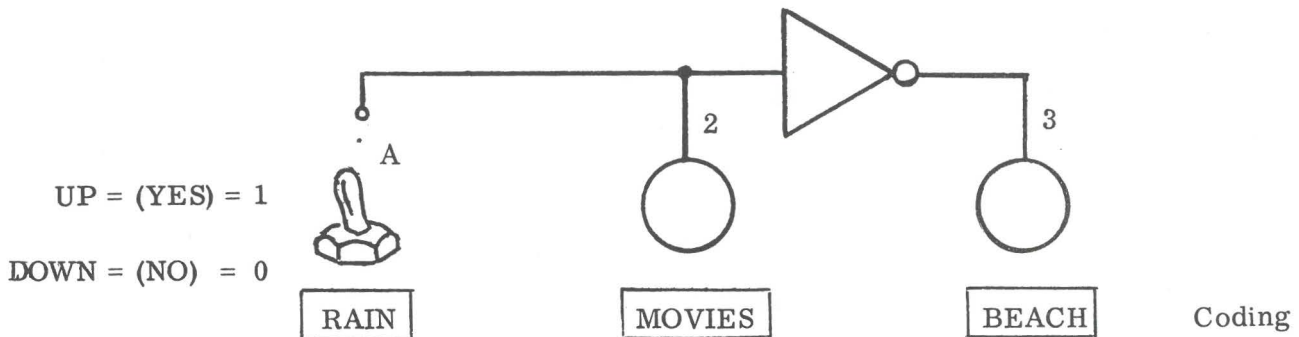
I am going to the MOVIES if it is RAINING.

I am going to the BEACH if it is NOT RAINING.

The Input Switch is coded for RAIN (UP is rain; YES). THE Lamps are coded for MOVIES and BEACH (ON is YES).

PROCEDURE:

Wire the Logic as shown. Enter the inputs and outputs in the table. Use the binary notations; 1 for YES and 0 for NO.



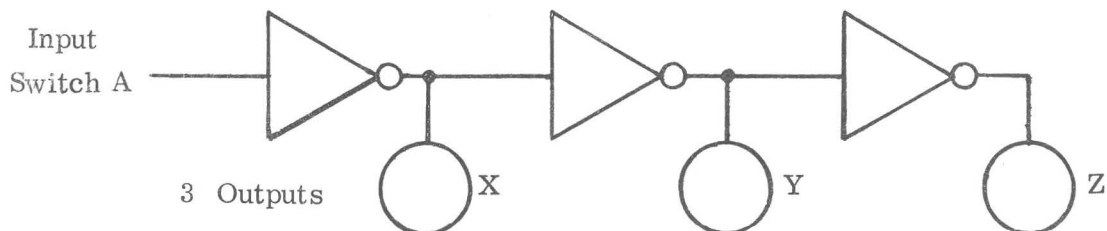
Experiment Logic 3 - 1 LOGICAL NOT PROBLEM

1 INPUT	2 OUTPUTS	
Switch A RAIN	Lamp 2 MOVIES	Lamp 3 BEACH

The electrical circuits for the logic elements are reviewed in experiment 75.

EXERCISES:

1. Prepare a TABLE for the following logic diagram.



2. Consider the problem; Press the gas pedal if the light is not red. Using a switch coded as LIGHT RED and a lamp coded as GAS PEDAL, sketch the diagram to solve this logic problem.

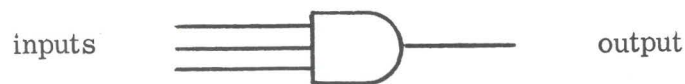
OBJECTIVE:

This experiment demonstrates the logical operation of the AND Digital Computer Logic Element. The TRUTH TABLE is also introduced.

DESCRIPTION:

I can go to the beach if it is a sunny day and I have a car. I will wear a raincoat if it is raining and I am going outside. I will buy new shoes if my old shoes are worn and I have the money. We will launch the rocket for the moon if the electrical system is operating and if the control system is functioning and if the power system is operating and if the oxygen system is working and if...etc... The result of the AND logical operation is a 1 when all of the input conditions are a 1. The first, and the second and the third inputs must be 1. This AND function is essential in everyday logic and also in the operation of Digital Computer Logic.

The symbol for the AND element is given below:



AND SYMBOL

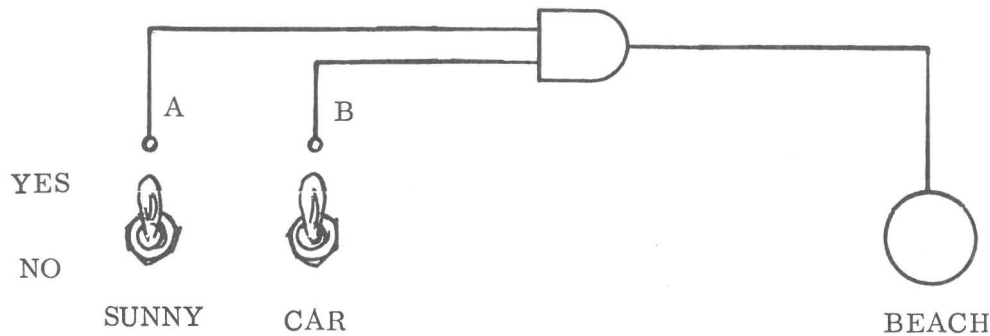
PROCEDURE:

Wire the Logic for the experiment shown on the following page.
The logic to be solved is:

I am going to the BEACH if it is a SUNNY day and I have a CAR.

We assume here that we need a car to go to the beach. The computer logic is only as good as our statement.

The logic for this experiment is shown below:



Experiment 4 - AND LOGIC PROBLEM

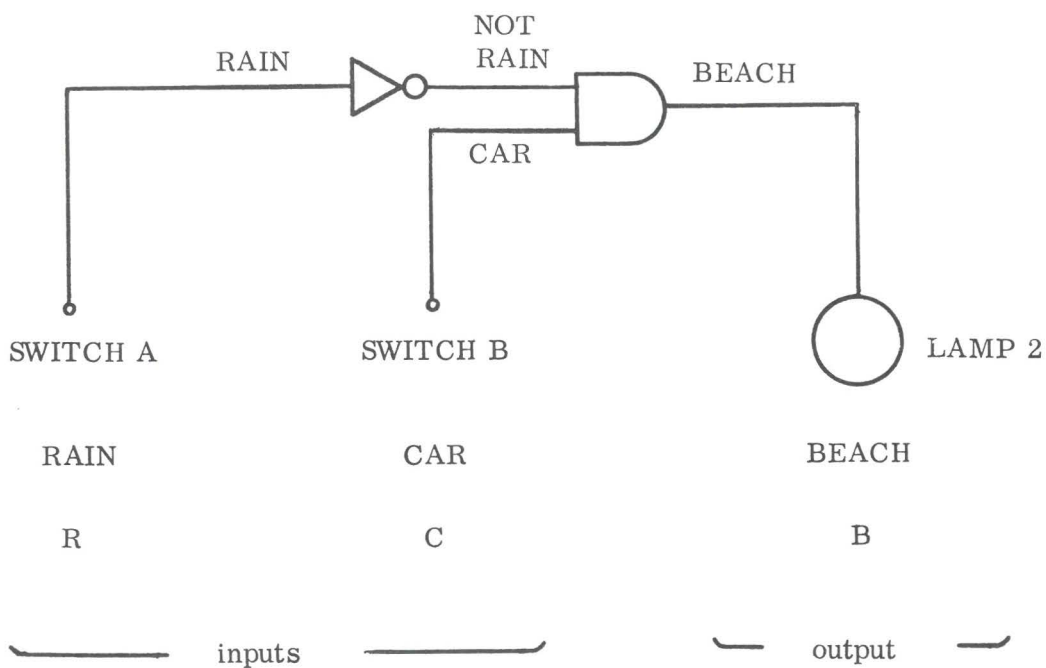
Operate the input switches and enter the results in the Table.
A YES is given by a 1 (Switch Up, Light ON) and a NO is given by a 0 (Switch Down, Light OFF). The following symbols are used: S means SUNNY DAY, B means go to the BEACH . The coded letters are used for convenience, and such shorthand coding is widely used in computer logic and elsewhere (to wit: FDR, LBJ, JFK, NASA, etc.) Define your own abbreviation for CAR . The table below shows the TRUE relationship between the input and output and is called a TRUTH TABLE.

TRUTH TABLE

SWITCHES		LAMPS
SUN	CAR	BEACH
S		B
0	0	
0	1	
1	0	
1	1	

QUESTIONS:

1. Prepare a Truth Table for 3 inputs to an AND gate. Inputs S, C and M, (Sunny day, Car, Money). Define the output as B (Beach). Write all conditions for the different inputs. There are 8 different combinations of inputs.
2. Wire this logic on the trainer. Set all inputs and check the tables. Are your results correct? _____
3. Prepare the Truth Table for the problem shown below. This logic problem combines the NOT with the AND.



The problem solved above is:

We go to the BEACH if it is NOT RAINING and we have a CAR.

Check the results on the trainer.

OBJECTIVE:

The purpose of this experiment is to demonstrate the operation of the Logical OR element.

DESCRIPTION:

I will wear my raincoat if it is snowing or if it is raining. I will stay home from school (work) if it is Saturday or if it is Sunday or if I am ill. The results of the OR Logical function is a 1 if any of the inputs are a 1 . The output is a 1 if the first input is a 1 or if the second input is a 1 or if the third input is a 1 . The symbol for the OR Logical Element is:



OR

LOGIC SYMBOL

PROCEDURE:

Wire the Logic for the experiment given on the following page..

The problem solved is:

I will wear my raincoat if it is raining OR if it is cold OR if I will be away for the night.

The coded abbreviations used are:

WR (wear raincoat) (lamp output), RN (it is raining), CD (it is cold), AN (away for the night). The coded letters may be written on a piece of paper and placed above the lamps and switches as shown in the figure.

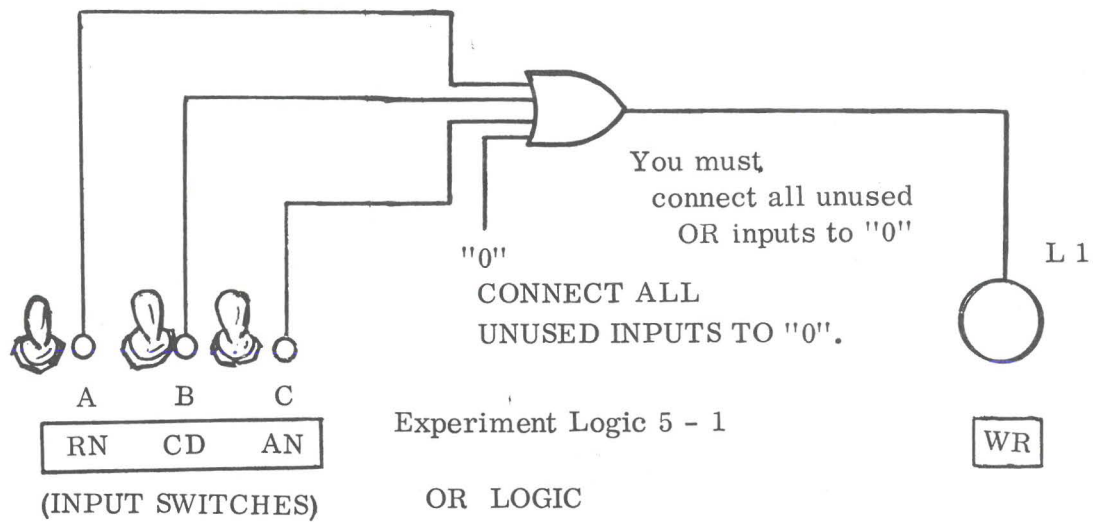


Table: Operate the input switches and record the output in the Truth

3 Inputs			1 Output
RN	CD	AN	WR
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

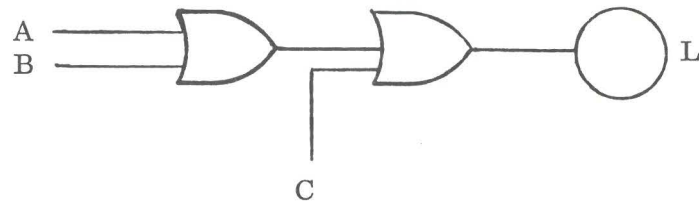
OR TRUTH TABLE

QUESTIONS:

1. Prepare a Truth Table for 4 inputs to an OR gate. How many different input combinations are there for 4 inputs? _____

2. State in words, what the OR Logic output is in relation to the inputs.

3. What is the Truth Table for the following logic.



OBJECTIVE:

This experiment illustrates how different logic elements can be "combined" with a number of inputs in a single logic system. Both the NOT element and the AND element are used. The interconnection of elements is given on a LOGIC DIAGRAM using the individual LOGIC SYMBOLS.

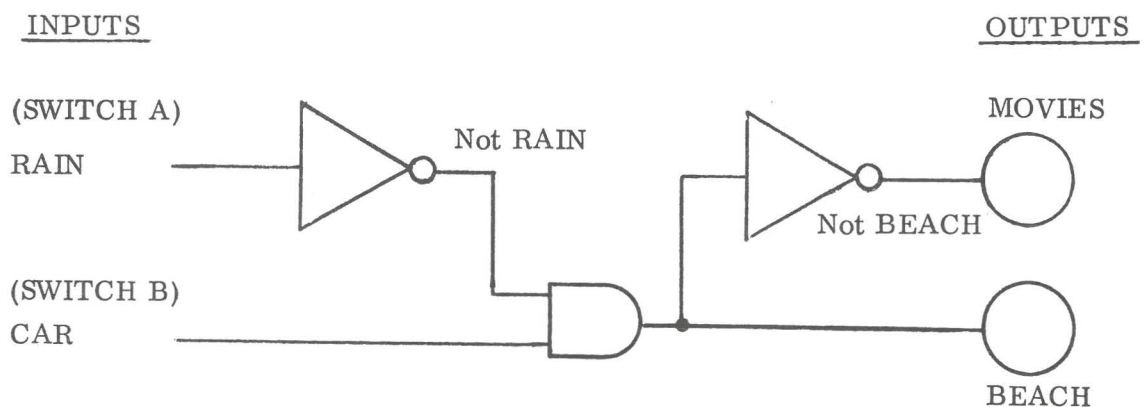
DESCRIPTION:

The system logic shown below includes both the NOT and the AND. The Logic Problem to be solved is:

I am going to the BEACH if it is NOT RAINING AND I have a CAR. I am going to the movies if I am NOT going to the beach. (We assume that we need a car to go to the beach and that we can walk to the movies.)

PROCEDURE:

Wire the Logic. Set up the Inputs and read the Outputs. Enter the Data in the Truth Table. Define your own two letter abbreviations for each input and output. Sketches are no longer used for input switches.



Experiment Logic 6 - 1

BEACH - MOVIES PROBLEM

LOGIC DIAGRAM

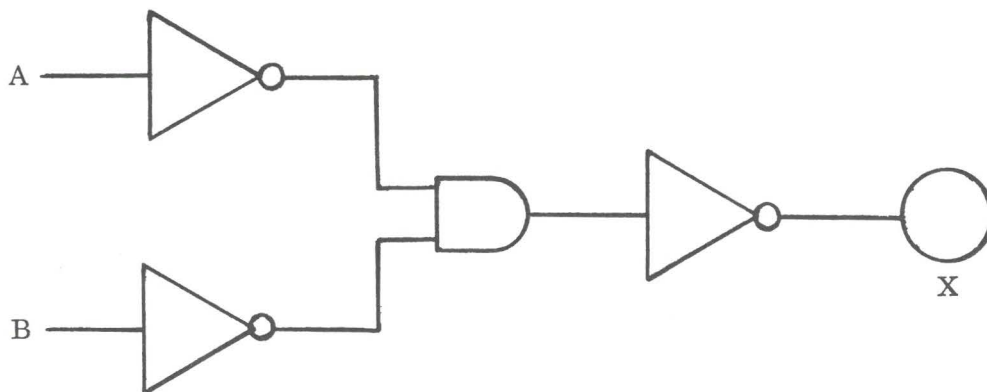
RAIN	CAR	BEACH	MOVIES
0	0		
0	1		
1	0		
1	1		

←SYMBOL

Run the Experiment. Is the Logic Correct? _____.

QUESTIONS:

1. Prepare the TRUTH TABLE for the following logic: Comment on this Logic Problem.



OBJECTIVE:

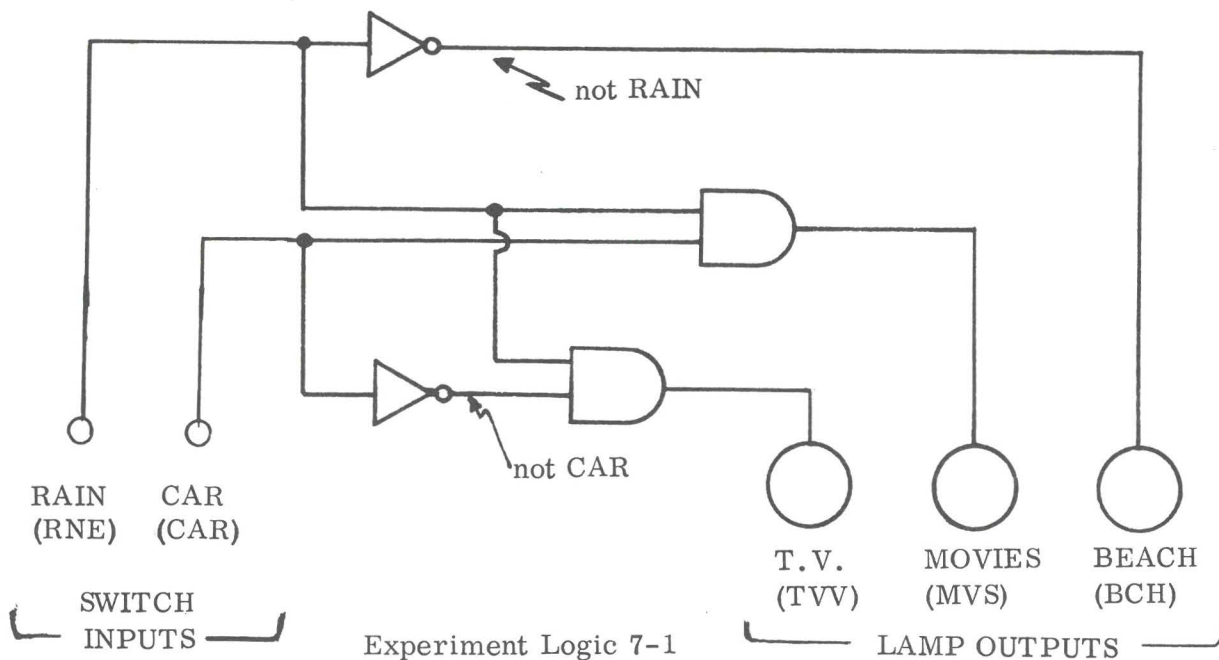
This experiment presents another example of "Combinatorial Logic" using the Not and the And. A more complex problem is presented.

DESCRIPTION:

The problem to be solved in this experiment is the reverse of the previous experiment. We live near the beach, but have to drive a car to go to the movies. The logic therefore is:

If it is not raining, I go to the beach (whether or not I have a car).
If it is raining and I have a car, I go to the movies. If it is raining and I do not have a car, I stay home and watch TV.

The Logic Diagram and the Truth Table are convenient ways of denoting the Logic to be implemented. It can become difficult to remember a complex logical expression and it is impossible to remember very long expressions. It is also interesting to note that the output result depends on the logic that we attribute to the situation ; i. e., whether we need the car for the movies or the beach (as in Experiment 6). The output from the logic is only as valid as the rules, or the wiring, or the "program" that we apply.

PROCEDURE:

Wire the Logic from the diagram in Figure 7-1. Operate the inputs and enter the results in a Truth Table below:

INPUTS		OUTPUTS		
RNE	CAR	TVV	MVS	BCH

Is the Logic correct? _____

QUESTIONS:

1. Draw the Logic Diagram for the following logic:
I will go to the beach if it is sunny and Not Sunday. I will watch TV if it is Sunday (Ballgame day). I will watch TV if it is not sunny. (Note that I will watch TV if it is Sunday OR if it is Not Sunny.)

The Inputs are SUNNY, SUNDAY.
The Outputs are BEACH, T.V.
Prepare the Truth Table.
Draw the Logic Diagram.

OBJECTIVE:

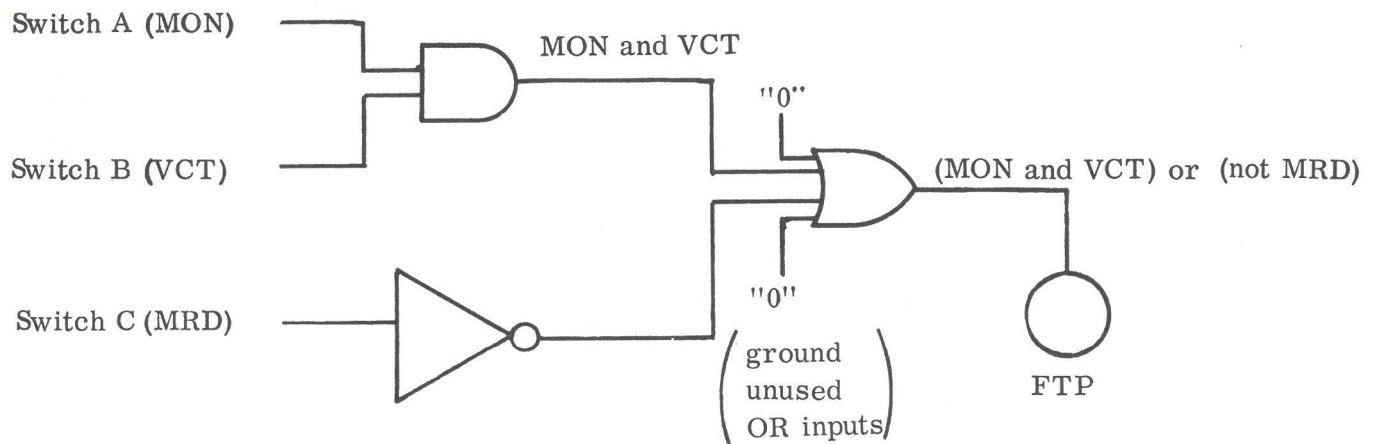
This experiment illustrates how LOGIC can be specified by a simple sentence or EXPRESSION or Statement or EQUATION. This EXPRESSION is equivalent to the Logic Diagram. It is interesting to note that we can prepare a logic situation or expression without actually knowing what it means.

DESCRIPTION:

Consider the following expression using abbreviations:

Lamp FTP is ON for (MON and VCT) or (not MRD)

The Logic Diagram is given in Figure 8-1 below:



Experiment Logic 8-1
LOGIC EXPRESSIONS

Another way to write the expression for an output is with an equals sign(=) as below:

$$\text{FTP} = (\text{MON and VCT}) \text{ or } (\text{not MRD})$$

A simple way of writing the logic sentences for the problem in experiment 7 is:

BEACH = NOT RAIN
MOVIES = RAIN AND CAR
TV = NOT CAR AND RAIN

PROCEDURE:

Wire the Logic as shown. Prepare a Truth Table (three inputs and one output). Set the Input Switches and record the Output Data.

Is the Logic Correct? _____.

QUESTIONS:

1. Prepare a Truth Table for the following expressions or equations.

$RNC = RNE \text{ or } SNE$ (Raincoat equals Rain or Snow)

$GLF = SUN \text{ and } WRM$ (Golf equals Sunday and Warm)

2. Draw the Logic Diagrams for the above two expressions.

3. Draw the Logic Diagrams for

$GLF = (SAT \text{ or } SUN) \text{ and } WRM$

4. Can you guess what the above expression (in question 3) means?

5. Can you guess what the experiment expression for FTP means ?

(You probably can not. The answer is in the answer manual).

OBJECTIVE

This experiment demonstrates a simple logic system where the output is a 1 when the Majority of the inputs are a 1. Such a system is often referred to as a Voting system. Similar logic is frequently used in computer systems.

DESCRIPTION

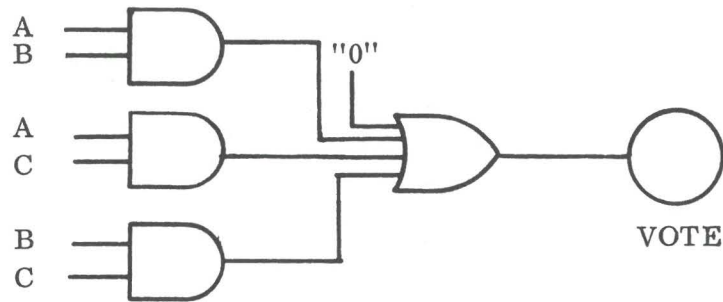
The Truth table for the Majority Logic system is given below:

A	B	C	VOTE
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

It is interesting to note that the Majority Circuit is a Carry Logic for a Binary Adder and is an Odd Parity Logic for code checking as will be discussed in later experiments.

PROCEDURE:

Wire the Logic and Check the Truth Table.



Experiment Logic 9-1

MAJORITY LOGIC

QUESTIONS

1. Prepare a Truth Table and draw the logic for a Minority Logic.
(The output is a 1 when the 1's are in the minority - or when the 0's are in the majority).
2. Draw the Logic for a complete voting system with 3 inputs. Use two output lamps PASS and FAIL to show the results of the vote.
3. Prepare a Truth Table for a voting system with 4 inputs and three outputs denoting PASS, FAIL, and TIE.
4. Draw the Logic as an optional exercise.

OBJECTIVE:

This experiment demonstrates the Logic required to detect when two binary inputs are identical.

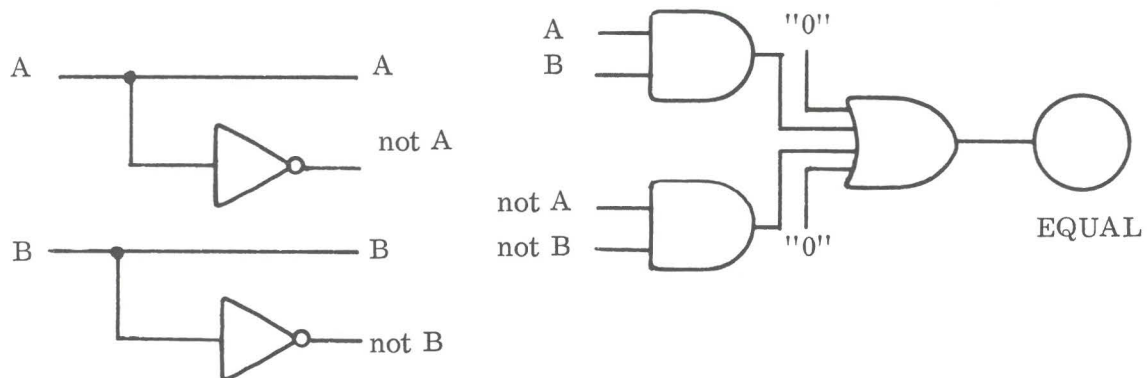
DESCRIPTION:

Consider two inputs A and B. They are identical when both are a 1 or when both are a 0.

TRUTH TABLE

A	B	EQUAL	Check
0	0	1	
0	1	0	
1	0	0	
1	1	1	

The Logic Diagram is shown below:



Experiment Logic 10 - 1

DIGITAL COMPARISON

PROCEDURE:

Wire the Logic. Set the Inputs through all combinations and record the outputs.

Are the outputs correct? _____

QUESTIONS:

1. Prepare the Truth Table for an INEQUALITY Logic.
2. Write the Logic Expression for the EQUAL Comparison Logic.

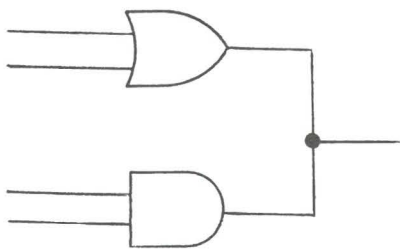
EQUAL =

OBJECTIVE:

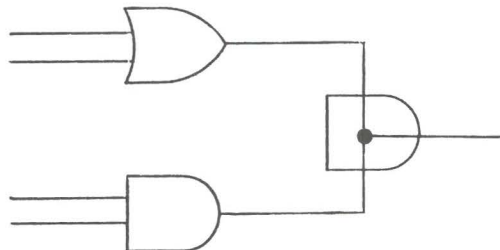
This experiment demonstrates the logic results when the outputs of various logic elements are connected together. This type of logic is referred to as Output Logic or Collector Logic (because the logic element outputs are from the "collector" terminals of transistors).

DESCRIPTION:

We have examined the results of each of the logic elements and have specified the outputs as a function of the inputs. What will happen if the outputs of two logic elements are connected to each other. There is no fixed answer. It all depends on the exact type and circuit design of the logic element used. In some systems, the connection of the outputs is not permitted and the logic will not work. (The TTL Logic is a system where connection of outputs is not permitted.) The trainer logic uses DTL elements and the connections of outputs are permitted. The output of any element is either a 1 or a 0. In DTL, the output is normally a 1 which is forced to a 0 by the internal electronic semiconductor. Therefore, when outputs are connected in parallel or together, a 0 on any of the logic outputs will force a 0 output. In order for a 1 output to occur, all the outputs must be a 1. This is the definition of an AND. Thus when outputs are connected together, they are really connected as an AND function. Two logic elements are shown connected below. The diagram on the right shows a symbol used to show that there is an AND function. There is no element added and the symbol is only used in the diagram to show the logic function resulting from an output connection and is helpful in understanding the logic.



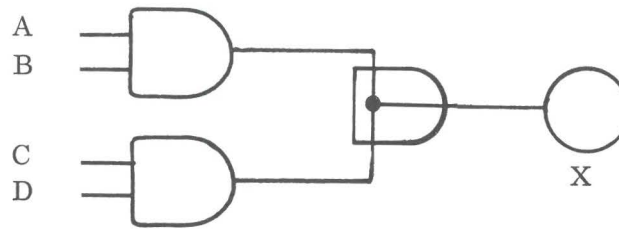
PARALLEL CONNECTION



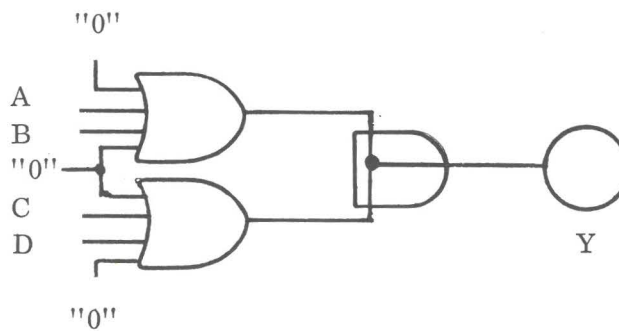
SYMBOL

PROCEDURE:

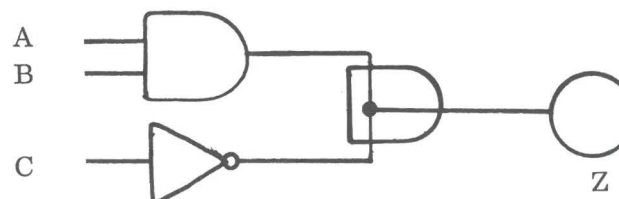
Wire each of the different Logic systems shown; one at a time.
Set up the switches, prepare the truth tables, and enter the results in the tables.
Does the output "collector" logic act as an AND Logic element? _____



Experiment Logic 11 - 1
AND - AND



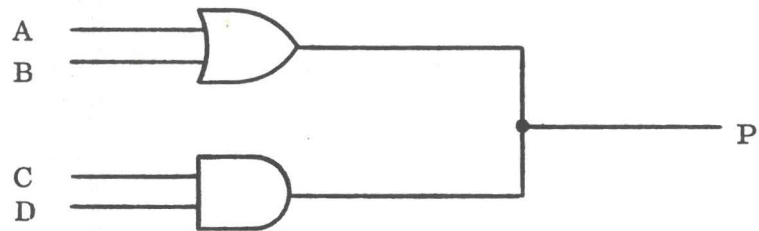
Experiment Logic 11 - 2
OR - AND



Experiment Logic 11 - 3
NOT - AND

QUESTIONS:

1. Draw the Truth Table for the following Logic Diagram.



OBJECTIVE:

This experiment describes the Logical function known as the Exclusive-Or. It is quite common and useful in computer circuits.

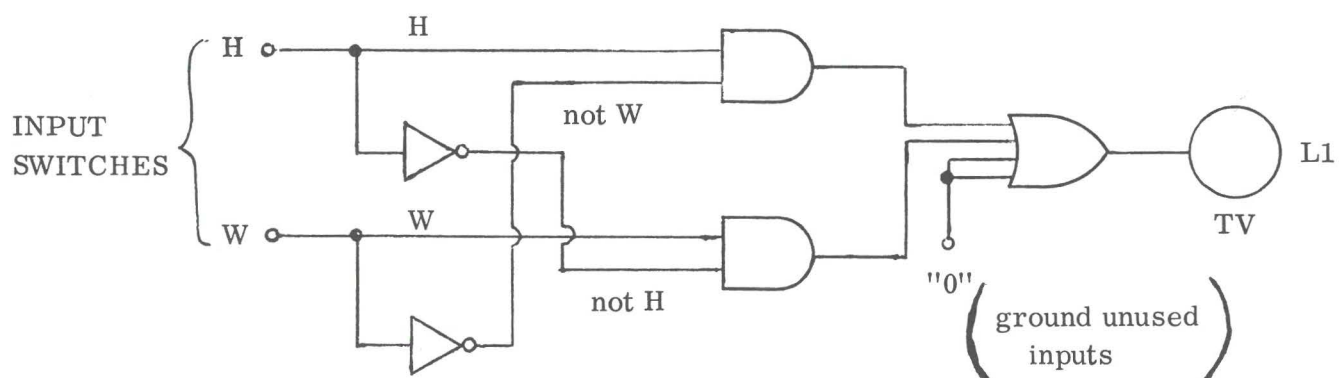
DESCRIPTION:

Consider the Logic of the following statement :

The television will be on when the Husband is in the house alone, or when his Wife is in the house alone. It will be off when they are both out and it will be off when they are both in the house, so that they may talk with one another. The logical expression is:

$$TV = H \text{ and not } W \text{ or } W \text{ and not } H$$

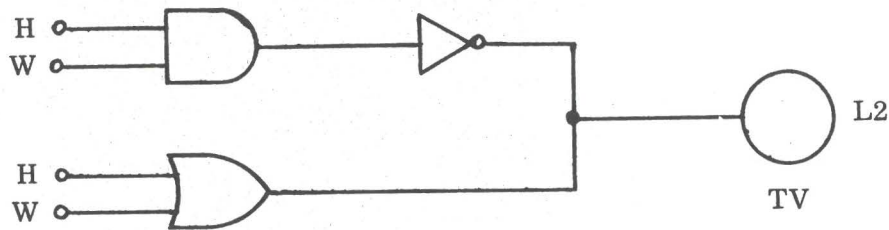
This function is known as the Exclusive - Or. The output is a 1 when one of the inputs is exclusively a 1. That is, the output is 1 when one but not both of the inputs is a 1. The normal or logic presents a 1 output when any number of the inputs is a 1 and is called an "Inclusive-Or" or simply an OR.

PROCEDURE:

Experiment Logic 12 - 1

NOT - AND - OR

Figure 12-2 shows the Exclusive - OR Logic using the Collector Logic described previously. Wire both Figure 12-1 and 12-2 logic circuits together and verify the performance of both logic systems. Enter both outputs in the Truth Table below.



Experiment Logic 12 - 2

COLLECTOR LOGIC

Logic 12-1 Logic 12-2 Figure 12-3

H	W	L1	L2	L3
0	0			
0	1			
1	0			
1	1			

A Logic Symbol for an Exclusive - OR is shown below:

Frequently, the Exclusive - OR is included in a single element and does not have to be implemented using a combination of elements. Wire this Element on the trainer and enter the results in the above table.



EXCLUSIVE - OR SYMBOL

Experiment Logic 12-3

QUESTIONS :

1. Consider a three input Exclusive - OR element where the output is a "1" only when a single one of the inputs is a "1". Prepare the Truth Table and Logic Diagram for this function.

OBJECTIVE:

This experiment demonstrates the logic required to compare two sets of inputs to determine if they are equal or identical. Comparisons are very important in digital logic systems and form the basis for DECISIONS in many operations.

DESCRIPTION:

Consider one set of input switches A B which can be set at 01 or 00 or 10 or 11. Consider another set of input switches which can be set in any combination and denoted as C D. We desire the output to be a 1 when both sets are identical. That is AB = 01 and CD = 01, or AB = 11 and CD = 11 or AB = 00 and CD = 00 or AB = 10 and CD = 10.

The Truth Table is given below:

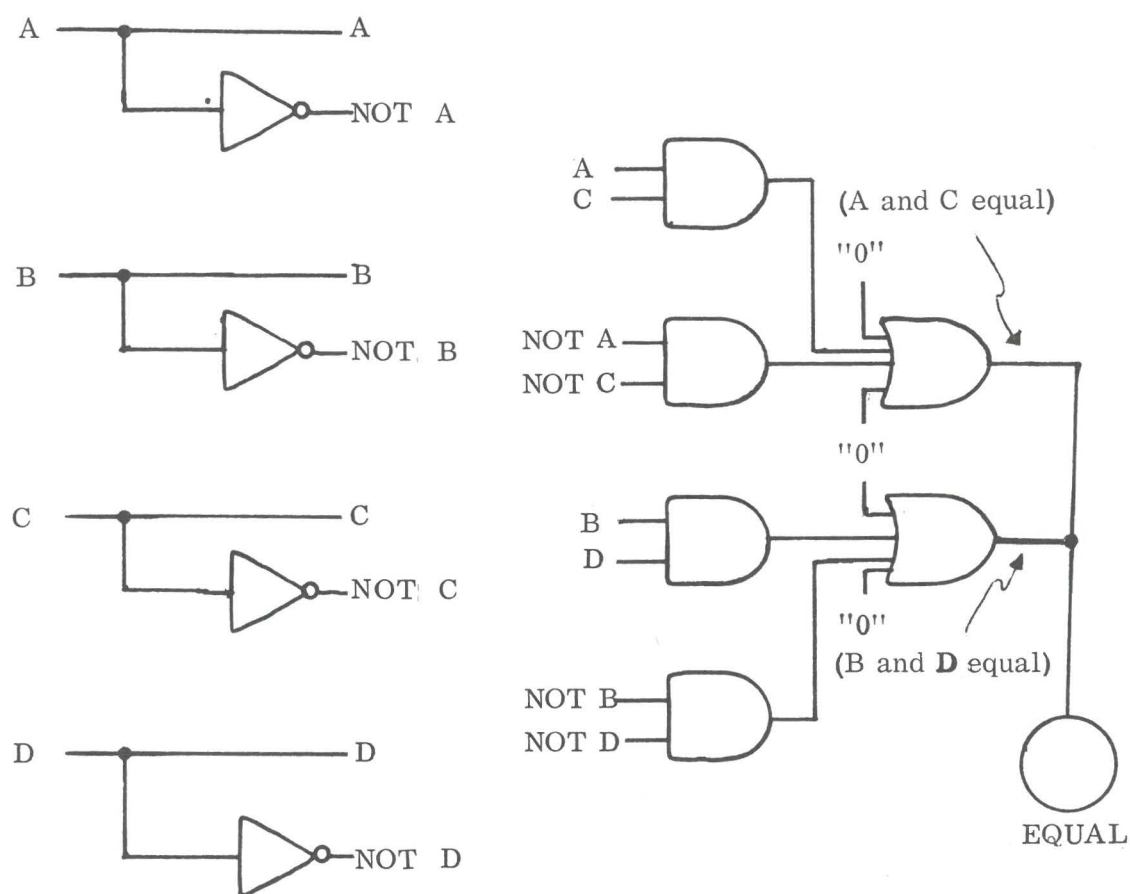
One-set of inputs		Another set of Inputs		
A	B	C	D	EQUAL
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

There are 16 conditions and four equality conditions.

The logic is implemented by comparing A to C and by comparing B to D.

PROCEDURE:

Wire the Logic shown in Figure 13-1.

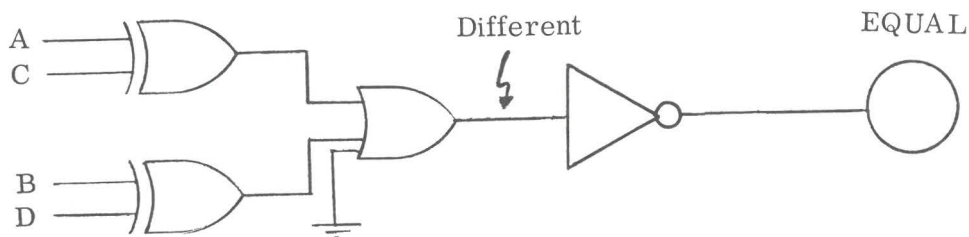


Experiment Logic 13-1
EQUALITY COMPARISON

Check the operation of the equality comparator by applying various combinations of switch inputs.

Does the Logic operate properly? _____

The Logic Diagram for the same logic using the EXCLUSIVE-OR ELEMENT is shown below. The output of an exclusive-or is 1 when the inputs are different. Therefore, the number is different if one comparison OR the other comparison are different.



Experiment 13-2
Exclusive-OR Comparison

Wire this logic.

Does the Logic Operate Properly? _____

QUESTIONS:

1. Can you think of one practical application of such a comparison for equality for two sets of inputs? _____
2. Write the Logic Expression for the above Equality Comparison.
3. Comment on the use of the Exclusive-OR.

OBJECTIVE:

The purpose of the experiment is to demonstrate the technique used in designing a Logic System from a statement of the Logical Problem.

DESCRIPTION:

The Logic of the situation is as follows:

If it is Sunday, I will watch the ballgame on TV. On any other day, if it is not raining and it is not cold, I will go to the beach. Except for Sunday, if it is cold and not raining, I will go to the city. However, if it is raining, I will watch TV.

PROCEDURE:

1. The first step is to prepare the TRUTH TABLE.

Hint: The inputs are: RAIN, COLD, SUNDAY
 The outputs are: BEACH, TV, CITY

Draw a completed Truth Table as a guide.

[illegible]

2. Write the Logical Expressions for each output:

BEACH =

TV =

CITY =

3. Draw the Logic Diagram.

4. Wire the logic and check the truth table.

Does your Logic work? _____

QUESTIONS:

1. Write the Logic Expressions for each output using only one letter for each item (B for beach , C for cold, etc..)
2. Draw the logic diagram to determine only the TV output.

OBJECTIVE:

The Truth Table, the Logic Diagram and Logic Expressions are some techniques for representing a Logic System. Another useful way is with Logic Maps or "Karnough Maps". This experiment describes the use of these maps.

DESCRIPTION:

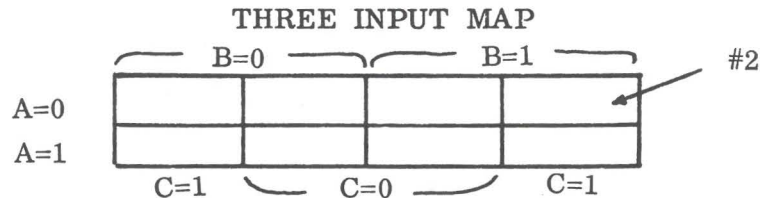
A Logic Map is shown in the figure below for a two input logic circuit (A and B are inputs and X is the output). The MAP is set up for the AND gate. Note that the box marked (Note 1) represents A = 0 and B = 1. The AND gate output for this condition is X = 0.

		<u>AND</u> MAP	
A Input {	A=0	X=0	(Note 1) X=0
	A=1	X=0	X=1
		B=0	B=1
		B Input	

The OR Gate Logic Map is shown below: Note that the symbols A Input, B Input, and X = are not used. In the OR gate, the only condition where X = 0 is where A = 0 and B = 0.

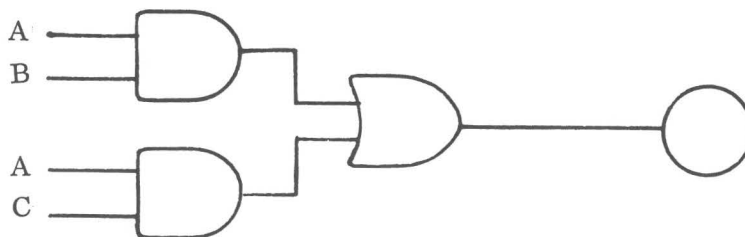
		<u>OR</u> MAP	
A=0		0	(Note) 1
A=1		1	1
		B=0	B=1

A three input logic map is shown below: Note that there are eight different boxes corresponding to the eight different conditions. The box noted by (#2) corresponds to $A = 0$, $B = 1$, $C = 1$.



PROCEDURE:

Wire the Logic shown in Figure 15-1.



Experiment Logic 15-1

LOGIC MAP CIRCUIT

Prepare the Logic Map for the logic in the above 3 input MAP. Check the operation of the Logic compared with the Logic Map. By observing the logic MAP, attempt to observe a simpler form of Logic Diagram.

QUESTIONS:

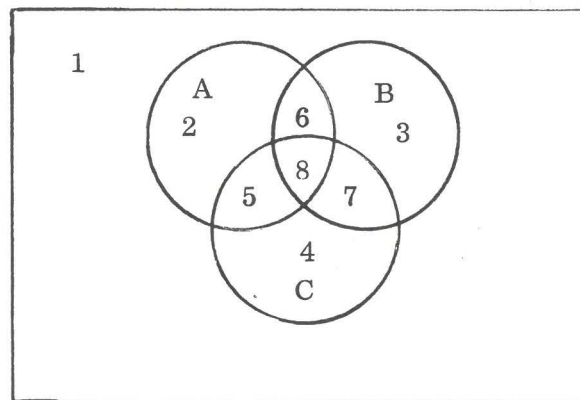
1. Draw a Logic Map for a four input system. Fill in the MAP with the Equality Comparison Logic of Experiment 13.

OBJECTIVE:

The Truth Table, Logic Diagram, Logic Expressions or Equations and Logic Maps can represent, or define, a Logic System. Still another way is by means of Venn Diagrams, also called Area Diagrams. This experiment illustrates the use of these diagrams. Venn Diagrams probably present the best or most "Visual" representation of Logic and is most useful in Logical Reasoning Theory.

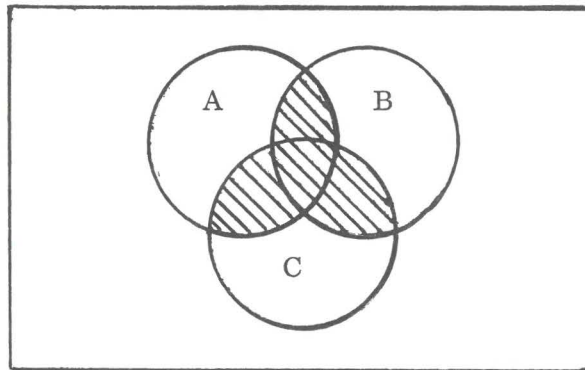
DESCRIPTION:

A Venn Diagram is shown below:



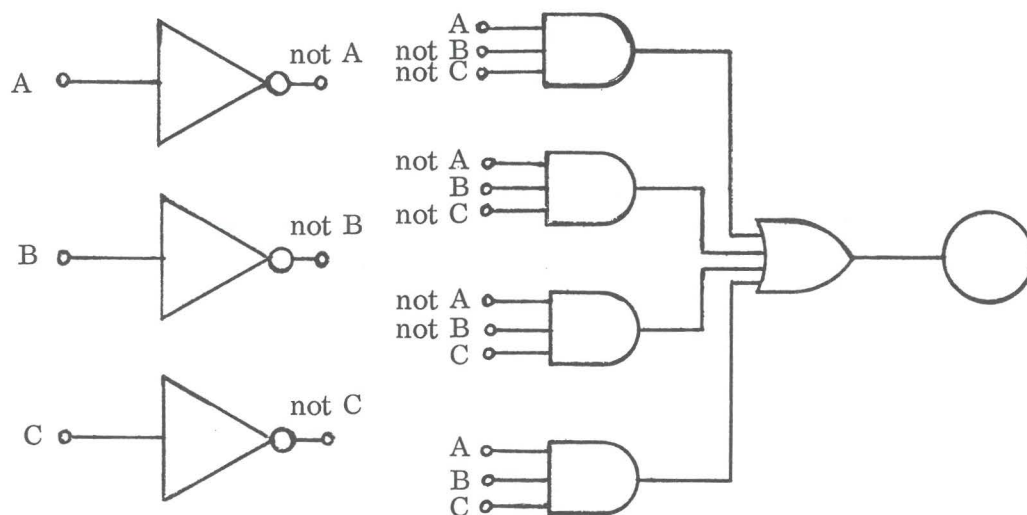
VENN DIAGRAM

Three circles are shown. The area within a circle represents a 1 input. Three circles are used to represent the inputs A, B, and C. Area (2) represents $A = 1$ only, i.e. $B = 0$ and $C = 0$. Area (6) represents $A = 1$ AND $B = 1$ AND $C = 0$. Area (7) represents $A = 0$ AND $B = 1$ AND $C = 1$. Area (1) represents neither A nor B nor C, i.e., $A = 0$, $B = 0$, $C = 0$. A particular logic system is represented by shading the Areas Corresponding to the output (X). The following figure is the Venn Diagram for the Majority Logic of a previous experiment.



MAJORITY LOGIC VENN DIAGRAM

The Logic for this experiment is ODD LOGIC. There are three inputs. The output X is 1 when there are an Odd number of input 1's, i.e., one 1 or three 1's. The Logic Diagram is shown in Figure 16-1. This Odd logic is quite useful to generate a SUM in addition and to generate a PARITY in code checks as will be discussed in later experiments.



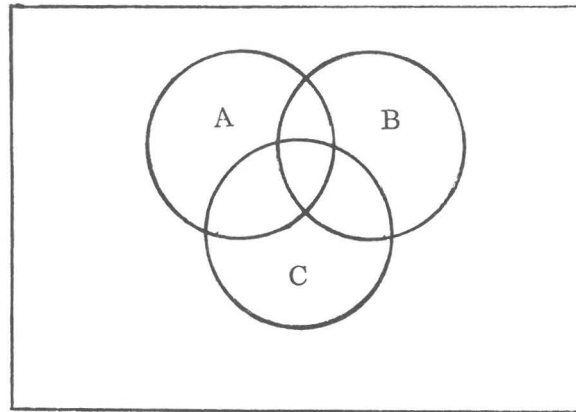
Experiment Logic 16 - 1

ODD LOGIC

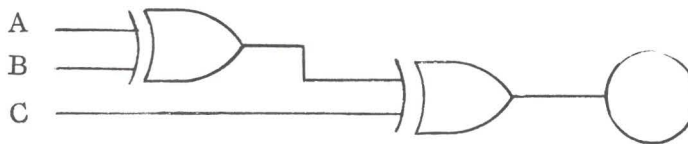
PROCEDURE:

Wire the Logic. Prepare the Venn Diagram for the Odd Logic.
Run the experiment and check the operation and the VENN

DIAGRAM.



Wire the equivalent Exclusive - OR
circuit below and check the operation.



QUESTIONS: Experiment Logic 16-2 EXCLUSIVE - OR

1. Draw the Logic Diagram for the Venn Diagram shown in Figure A.
2. Draw the Venn Diagram for the Logic in Experiment 6 (Combination Logic).
3. Draw the Venn Diagram for the Logic in Experiment 12 (Exclusive - Or).
4. Prepare the Venn Diagram for a three input AND, for a three input OR, for a NOT (Inverter).
5. Draw the Venn Diagram for the Logic of Experiment 15.

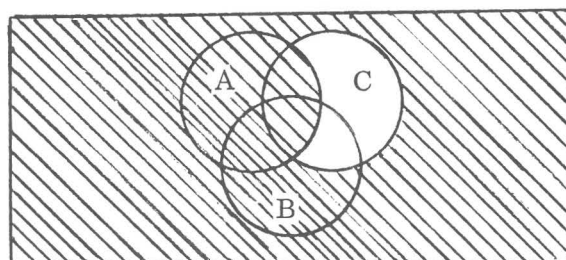


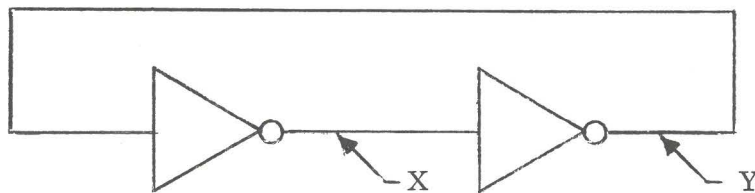
FIGURE A

OBJECTIVE:

This experiment illustrates how computer logic elements can be used to "remember" information and thus serve as a MEMORY element for storing digital or computer data.

DESCRIPTION:

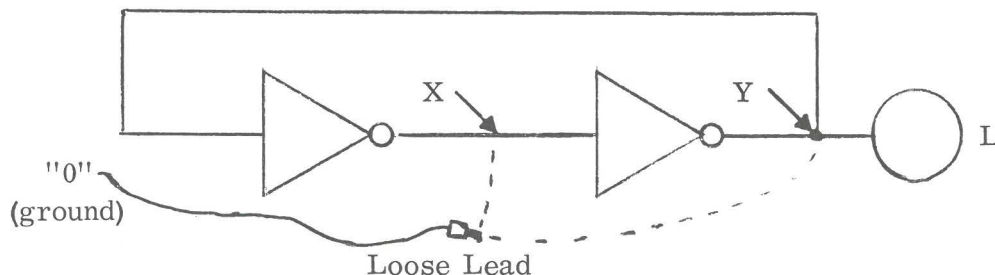
The logic elements shown below are two NOT elements or Inverters connected in a feedback path. The output of each one connects to the input of the next one. When the symbol at X is a 1, Y is a 0. Y is the input to the Inverter which has a 1 as the output. Thus X is 1 and Y is 0. When X is a 0, the output Y is a 1. There are thus two "stable" conditions and the electronics remembers or stores or memorizes one 1 or 0, yes or no condition. Inputs are normally applied to the computer by the switches. When the inputs are removed, the computer or logic trainer must remember the input data. This logic circuit is called a FLIP-FLOP.



MEMORY

PROCEDURE:

Wire the logic memory shown below in Figure 17-1.



Experiment Logic 17 - 1

NOT MEMORY LOGIC

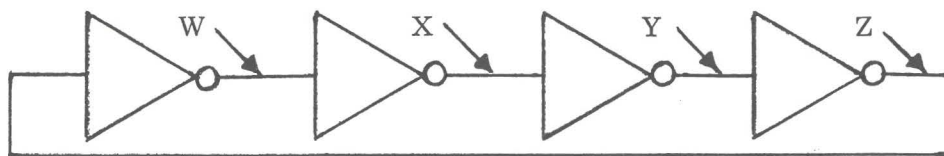
Record the output condition (L). Connect a loose lead to the "0" or ground bus line. Momentarily "touch" the X output, and then the Y output. Observe the the circuit remembers which output was touched last. Note that when the "0" is connected to an output, it forces a "0" on that output.

Repeat the tests a number of times and enter the data in the table.

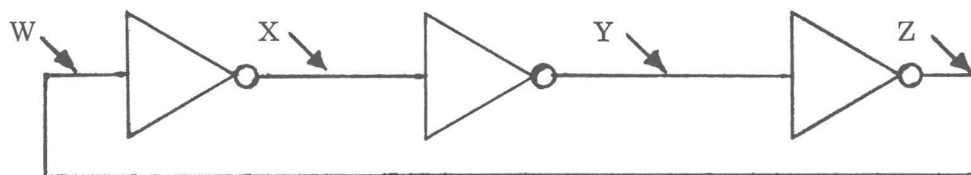
INPUT X or Y	L

EXERCISE:

1. What is the performance of the Logic shown below in (a) and in (b).



Logic (a)



Logic (b)

OBJECTIVE:

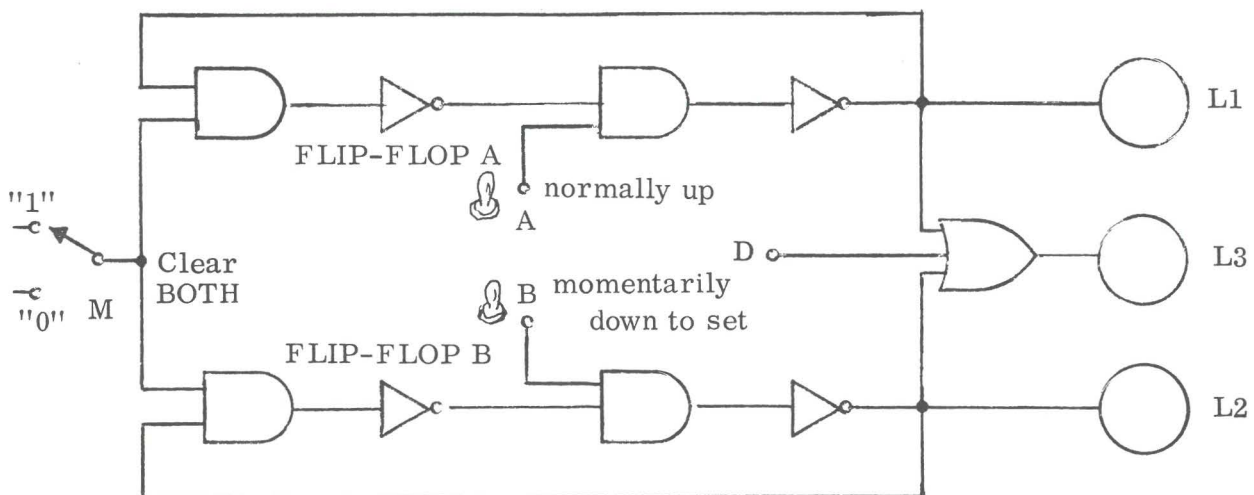
This experiment demonstrates how a LOGIC MEMORY is used in conjunction with logic elements to generate a Logical Output based on a SEQUENCE of inputs.

DESCRIPTION:

The previous experiment demonstrated how logic elements (NOT) can be used for memory. Figure 18-1 shows the logic for this experiment which combines a MEMORY and LOGIC utilization. The AND gates are used to enable "clearing" of the memory and "setting" of the memory. Switches A and B are normally Up and the inputs to the AND gates are thus 1. When any input is momentarily placed Down, the input to the AND gate is forced to 0 and the output of the following NOT is thus forced to a 1. Switch M is used to Clear the two position memory. Switch A is used to set the first memory element, and Switch B is used to set the second memory element.

PROCEDURE:

Wire the Logic given below:



Experiment Logic 18 - 1

SEQUENTIAL LOGIC

Place all switches Up. Momentarily depress switch M (down) to clear the memory (both lights off) and then place switch M Up. to enter data, momentarily depress (down) switch A and/or switch B (and then place switches UP).

Enter the following codes into the memory (clear and set) :

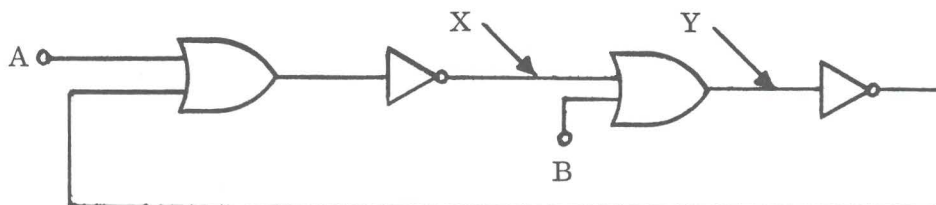
A	B
0	0
0	1
1	0
1	1

Output X is the OR output for inputs A, B, D. Note that the A and B inputs are from the memory. and D is from the input switch. The output thus not only depends on the present inputs, but on the previous inputs. Such a logic system is referred to as "sequential" logic. The output depends on the sequence of previous and present inputs.

Operate memory inputs A and B and M, and switch input D and observe the outputs. Make a Truth Table for various sequences used.

QUESTIONS:

1. Describe the operation of the following Memory element using the OR and the NOT elements. Note that the normal position for all inputs is 0 (switches down).



OBJECTIVE:

This experiment describes a means of shorthand notations on logic symbols and diagrams which results in simpler diagrams, but must be interpreted by the user to correspond to the logic elements that he has available. Algebra equation symbols can be substituted for "expressions" to yield Logic Algebra Equations or Boolean Algebra Equations.

DESCRIPTION:

The circle $\bigcirc$ on a logic element represents the Inversion or the NOT of the inputs. Figure 19-1 shows the NOT element. The not function could also have been represented by the symbol shown in Figure 19-2 with the symbol at the input.

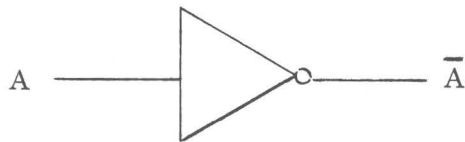


Figure 19-1

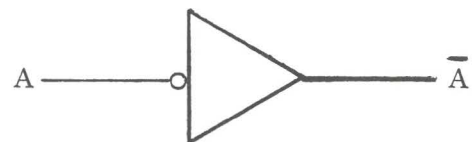


Figure 19-2

Consider Figure 19-3. The AND gate has two inverted inputs. Figure 19-4 shows how this logic is to be implemented. Figure 19-5 shows a logic system with NOT C as an input and an inversion on B as an input.

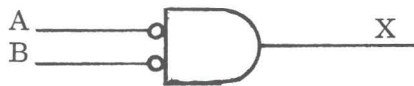


Figure 19-3

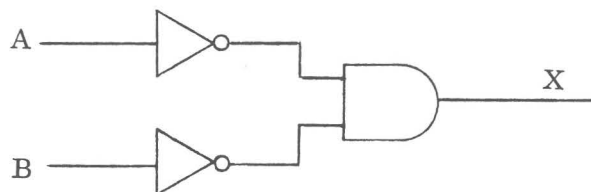
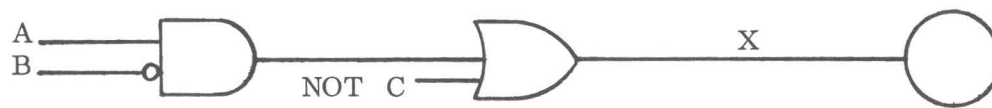


Figure 19-4



Experiment Logic 19-5

LOGIC DIAGRAM

PROCEDURE:

This experiment uses the logic of Figure 19-5. First draw a logic diagram corresponding to the Figure 19-5, but using the available elements. Predict the output Truth Table or Venn Diagram or Karnaugh Map from Figure 19-5. Run all possible combinations and check the results.

[illegible]

QUESTIONS:

1. Prepare the Truth Tables and the Logic Diagrams (using AND, OR, and NOT) for Figures 19-6, 19-7, 19-8, 19-9.



Figure 19-6

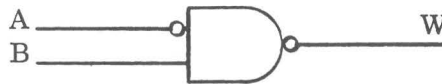


Figure 19-7

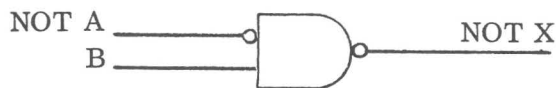


Figure 19-8

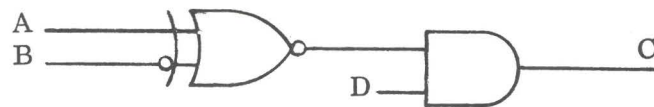


Figure 19-9

2. Prepare the Logic Expressions for each of the above diagrams
Use a BAR over a symbol to represent a NOT.

Thus Not C is written as $\overline{C}$

Use a plus sign + to represent the OR

Thus A OR B is written as $A + B$

Use a dot • to represent an AND

Thus A AND B is written as $A \cdot B$

The Logic Equation for this experiment of Figure 19-5

$$\text{is: } X = A \cdot \overline{B} + \overline{C}$$

3. Prepare the Truth Tables for the Logic Diagrams in Figures 19-10 and 19-11. Figure 19-10 is NOT-AND and is known as a NAND element. Figure 19-11 is NOT-OR and is known as a NOR element.



Figure 19-10

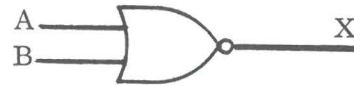


Figure 19-11

4. Prepare the Truth Tables for the Logic Diagrams in Figures 19-12 and 19-13. Comment on these elements.



Figure 19-12

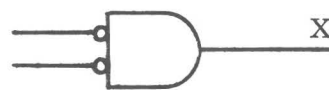
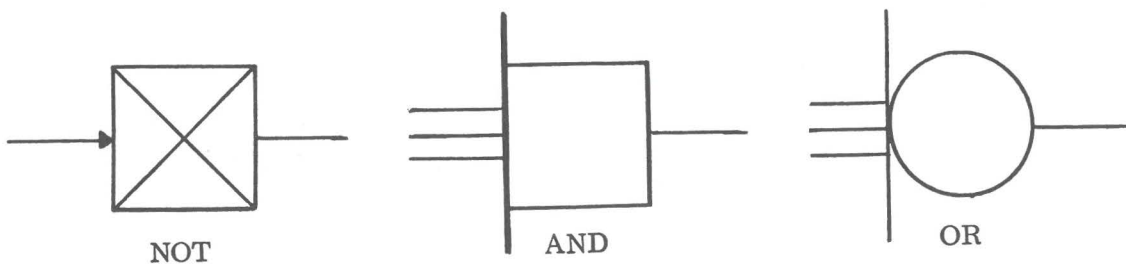


Figure 19-13

5. Equivalent logic symbols used in Electricity and Motor Control diagrams are shown below:



OBJECTIVE:

The previous logic problems were relatively simple and in most cases, we could have arrived at the answer with speeds comparable to that of the computer—or even faster. This experiment presents a problem that is much more difficult to solve and requires some logical support of a Computer System.

DESCRIPTION:

Consider the old puzzle or problem:

We are on an island which has two native tribes. Both tribes dress the same way. The difference is that one tribe always tells the truth and the other tribe always tells a lie. We meet two natives and must get important directions so it is necessary to find out if one or both of them are liars. We can't just ask, because both will say that they tell the truth. They will not answer irrelevant questions like, "is it day or night?". What questions must we ask to find out who tells the TRUTH and who is the LIAR?

The student can now attempt to solve this problem before proceeding further. The aim, of course, is to devise questions and answers and logic to solve this problem. Do not spend too much time trying to solve this difficult problem. Continue and read the descriptions.

SOLUTION: The first task, of course, is to set up a Truth Table representing the situation. There are four possible combinations as shown in the Table below:

NATIVE 1	NATIVE 2
L	L
L	T
T	L
T	T

There are 4 combinations. Using two switches, one also obtains four combinations as shown in previous experiments. This can lead us to believe that 2 yes or no (1 or 0) questions and answers will suffice to identify the four conditions and hence the natives. It is now necessary to formulate the two questions.

The computer logic may provide a rapid answer, but it is up to us to determine the questions or switch inputs and the logic or to "program" the computer with the logical rules and connections.

The student may wish to formulate the questions and the logic before proceeding. The questions and the logic are given on the following page. Do not waste too much time trying to devise questions. Continue to the solution.

QUESTION 1	QUESTION 2

QUESTIONS:

Both questions are asked of Native 1. (This is one possible solution.)

Question A. Is your companion a Truth Native ?
Question B. Are you both from the same tribe?

If the answers are: NO, YES, then who is the liar?
If your answer takes longer than 1 second, the computer can do it faster.

The complete Truth Table for the conditions and for the answers to the questions is given below:

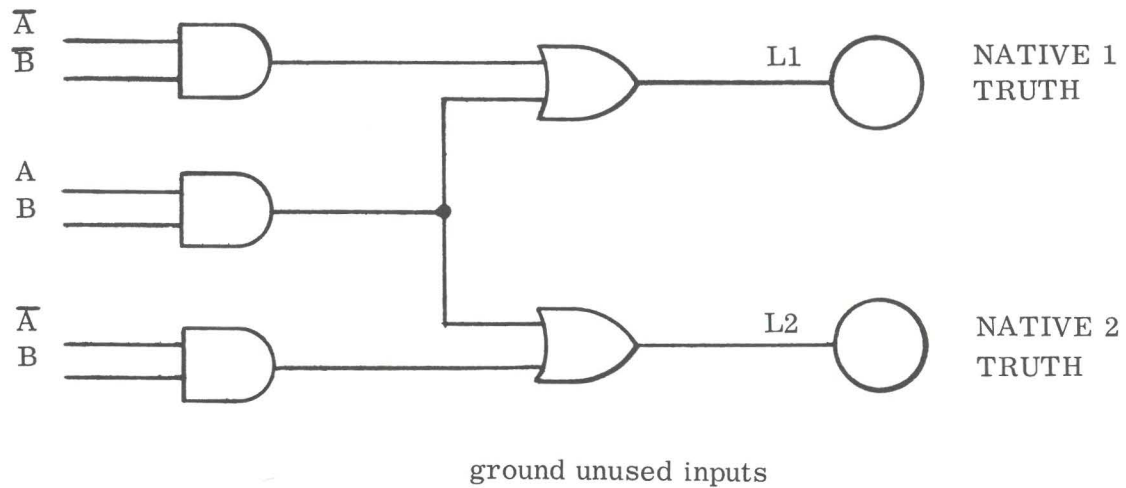
OUTPUTS		INPUTS	
NATIVE 1	NATIVE 2	ANSWER QUESTION A	ANSWER QUESTION B
0	0	1	0
0	1	0	1
1	0	0	0
1	1	1	1

1 = TRUTH
0 = LIAR

1 = YES
0 = NO

Native 1 is "Truth" when not A and not B or A and B
Native 2 is "Truth" when not A and B or A and B

The Logic Diagram is shown in Figure 20-1.



Experiment Logic 20 - 1
TRUTH - LIAR PROBLEM

PROCEDURE:

Wire the Logic. Select a combination of answers. Use the computer to test the "truthfulness" of the natives. Enter the data in a truth table. Check the accuracy of the results. Time the response time to get an answer from the computer. Compare this to the time it takes to figure out the answer "in your head" without referring to the Truth Table.

QUESTIONS:

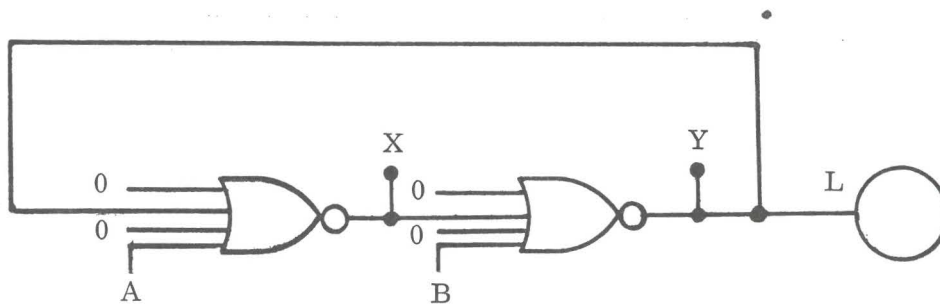
4. Draw the equivalent NAND system.
5. Draw the equivalent AND, OR, NOT system.

DESCRIPTION (FLIP-FLOP):

Two NORs are connected to each other. A and B are normally "0". When X=1, then Y=0 (and X=1). When X=0 then Y=1 (and X=0). There are thus two "stable-states". When A is switched up (1) X becomes 0 and Y becomes 1 and the output is L=1. (LAMP is ON). When A is switched down, the lite will remain ON. When B is switched UP, Y becomes 0; the lamp is OFF (L=0) and X becomes "1". When B is switched down, the lamp remains OFF. This circuit thus "remembers" the last switch which was set UP.

PROCEDURE:

Wire the experiment and set the switches as shown in the table. Verify that the lite is ON after A is switched UP and OFF after B is switched UP.



Experiment Logic 25-3
NOR MEMORY

QUESTIONS:

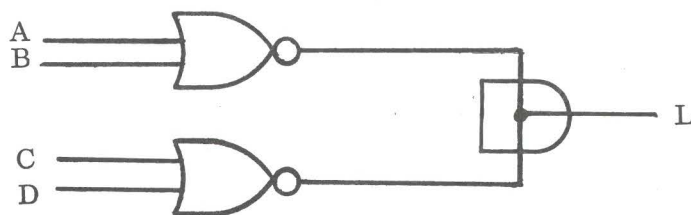
6. What is the operation if the first NOR has two inputs, A and C, and the second NOR has 2 inputs, B and D?

QUESTIONS:

1. How many different input combinations are possible with the 5 switches?
2. How many different combinations are possible with the full 7 inputs?
3. Draw the logic for a 16 input NOR gate.

DESCRIPTION (Parallel NORs):

When either NOR output is "0" or ground, the output is "0". The equation is the AND of the two outputs. This connection is drawn as shown below.



The logic equation is

$$L = \overline{(A+B)} \cdot \overline{(C+D)}.$$

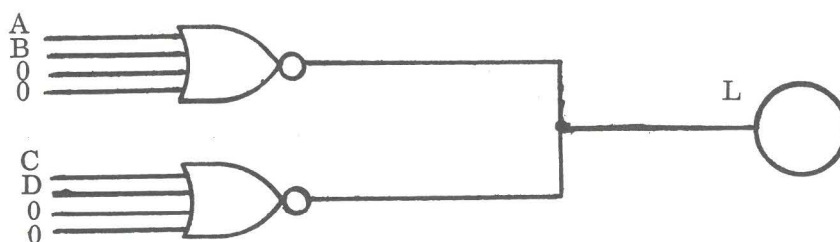
This can also be written (using DeMorgan's Theorem) as:

$$L = \overline{(A+B)+(C+D)} = \overline{(A+B+C+D)}$$

Parallel NOR's are thus equivalent to one multiple input NOR. This is then another way to implement an EXPANDER.

PROCEDURE:

Draw a truth table with A, B, C, D, & L. Wire the logic and check the outputs for various combinations.



Experiment Logic 25 - 2

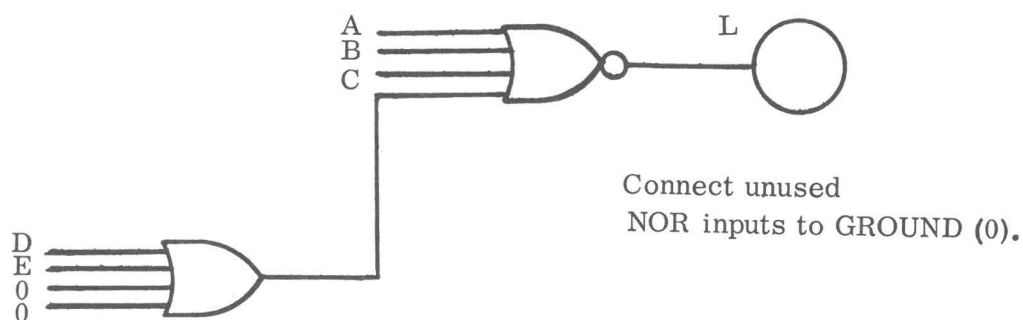
PARALLEL NORs

OBJECTIVE:

The NOR gate on the unit has 4 inputs. An EXPANDER is used to increase the number of inputs to a NOR. PARALLEL outputs provide an AND function. Two NORs can be used in a MEMORY flip-flop configuration.

DESCRIPTION (EXPANDER):

The NOR is an OR followed by a NOT. An OR gate is used as an EXPANDER on one of the NOR inputs as shown below to yield a 7 input NOR gate.



Experiment Logic 25-1

NOR / OR EXPANDER

PROCEDURE:

Wire the experiment. Set the input switches and enter the output in the table. Select other combinations to complete the table.

TABLE 25 - 1

A	B	C	D	E	L
0	0	0	0	0	
0	0	0	0	1	
0	0	0	1	0	
0	0	1	0	0	
0	1	0	0	0	
1	0	0	0	0	
1	0	1	0	0	

Setup the input switches and complete the truth table for each logic function. Check the outputs to see that the appropriate logic functions were performed.

QUESTIONS:

3. Draw a diagram for NOR logic system which generates the function $L = \overline{A}\overline{B} + \overline{C}$. (2) Draw the logic MAP for this function.

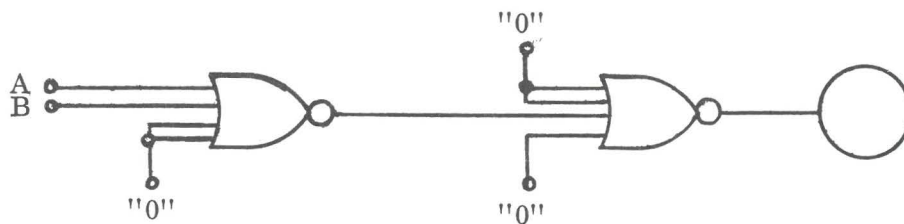
A	B	L OR	L NOT	L AND	✓
0	0				
0	1				
1	0				
1	1				

DESCRIPTION:

The NOR output is an inverted OR. By inverting the output of the NOR again, the result is once again the OR. The NOT gate function is provided by using the NOR gate with a single input. The AND gate function is provided by inverting the inputs.

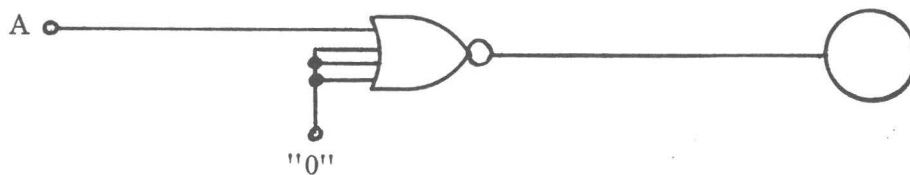
PROCEDURE:

Wire the experiment for each logic function below, (one at a time).



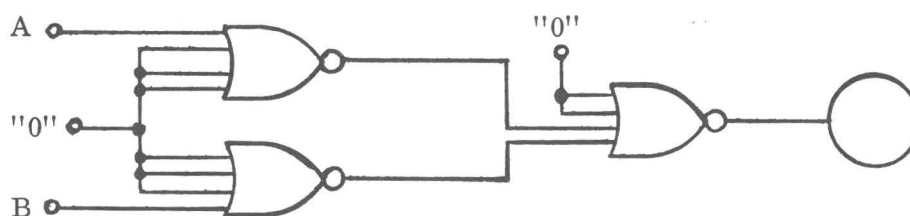
Experiment Logic 24 - 2

OR



Experiment Logic 24 - 3

NOT

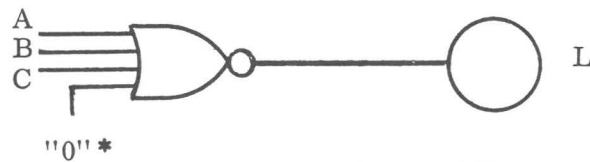


Experiment Logic 24 - 4

AND

TRUTH TABLE -NOR

A	B	C	L(1)	L(2)	✓
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			



* IMPORTANT : All unused NOR inputs must be connected to "0" or ground.

Experiment Logic 24 - 1

NOR LOGIC ELEMENT

A typical word application for the NOR is :

PLAY GOLF = NOT (RAIN or SNOW or VERY COLD)

QUESTIONS:

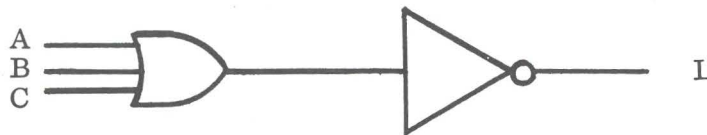
1. Which input condition (A, B, C) is the only condition which results in a "1" output?
2. Draw the Logic MAP and VENN Diagram for the NOR circuit.

OBJECTIVE:

This experiment introduces the NOR Logic Circuit.

DESCRIPTION: The equation for the NOR Logic Circuit is $L = \overline{A + B + C}$

It is read as "L equals NOT A OR B OR C. The result is the same as an OR circuit followed by a NOT circuit as shown below:



OR - NOT

NOR is short for NOT-OR. The symbol is shown below and is the OR symbol with a circle at the output. This circle represents an inversion or a NOT.



NOR

PROCEDURE:

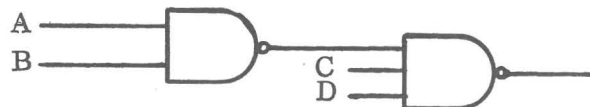
The TRUTH TABLE is given for the NOR circuit. First fill in column (1) with the expected NOR output. Then wire the experiment and enter the output data in column (2) and check both results.

Prepare the truth table in the DATA section and fill in the 8 conditions for A, B, C, and the expected results for L (column 1). Wire the experiment and enter L (in column 2) in the table. Check the results.

A	B	C	L(1)	L(2)

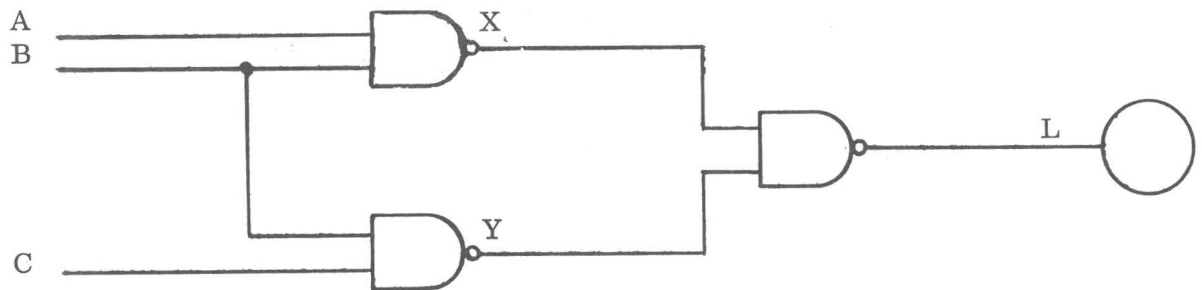
QUESTIONS:

2. Draw an all NOR circuit that performs the same function.
3. Draw an AND, OR, NOT circuit that performs the same function.
4. Draw the VENN diagram and Karnough Maps for this system.
5. Draw a NAND Flip - Flop Memory.
6. What is the Logic for a series NAND System.



DESCRIPTION (NAND LOGIC SYSTEM):

The diagram below uses 3 NANDs with 3 inputs (A, B, C).
Intermediate logic signals are X and Y.



Experiment Logic 23 - 2
NAND SYSTEM

PROCEDURE:

Write the equation for X and Y.

X =

Y =

Write the equation for L.

L =

PROCEDURE:

Wire the logic given above. Set - up the inputs given in the data table. The expected results are given in the table. Check the outputs. Select other inputs and enter the inputs and outputs in the data table.

A	B	C	D	L(1)	L
0	1	1	0	0	
1	1	0	1	1	
0	1	1	1	1	
1	1	1	1	1	
0	1	0	0	0	
1	1	0	0		

QUESTIONS:

Draw the logic for a quadruple AND - OR logic, i.e. for

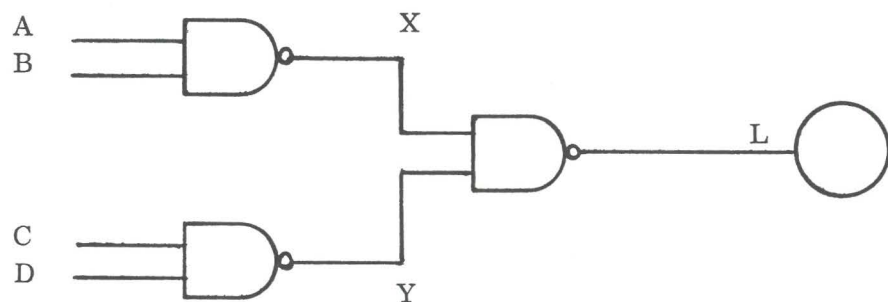
$$L = AB + CD + EF + GH.$$

OBJECTIVE:

This experiment demonstrates the most common and useful interconnection of NAND elements to perform the AND - OR logic function.

DESCRIPTION (AND - OR):

The logic diagram below performs the AND - OR function.



Experiment Logic 23 - 1
AND - OR LOGIC

The outputs X and Y for the NAND elements are given by the equations:

$$X = \overline{AB}$$

$$Y = \overline{CD}$$

The output L is given by:

$$L = \overline{XY}$$

$$L = \overline{\overline{AB} \cdot \overline{CD}}$$

Using DeMorgan's Theorem (Reviewed in Experiment No. 29)

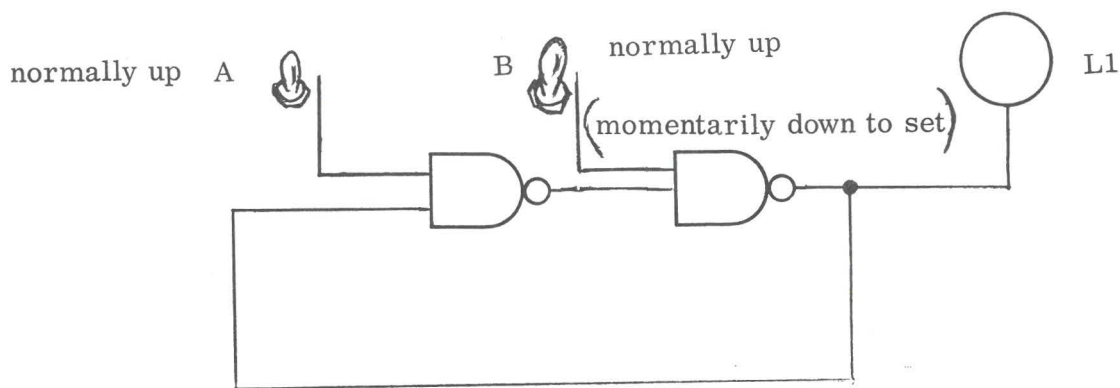
$$L = AB + CD$$

QUESTIONS:

4. How many input combinations are there for a seven input NAND element?
5. What is the maximum number of NAND inputs that can be derived from the four input NAND gate and four of the four input AND gates?
6. Would it be possible to increase the number of NAND inputs available using collector logic?

NAND FLIP-FLOP

Two NAND can be connected as a FLIP-FLOP MEMORY as shown below:



Experiment Logic 21-4
NAND FLIP-FLOP

Wire the above NAND Flip-Flop Logic. Set Switches A UP and B UP. Set switch B DOWN then UP to Set L1 ON. Set switch A DOWN then UP to Clear L1 OFF. Check the operation. Enter the Data Below:

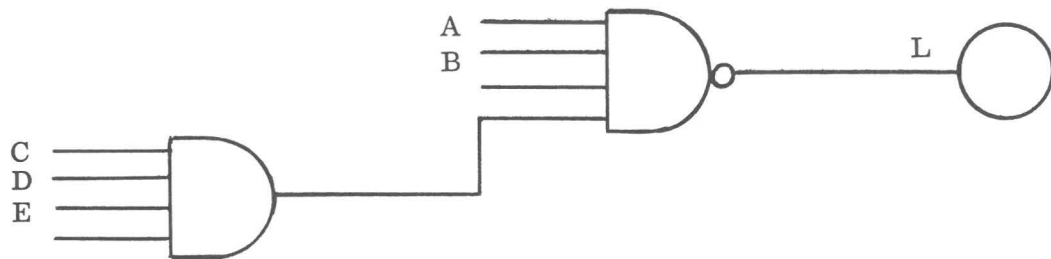
<u>FLIP-FLOP INPUT</u>	<u>OUTPUT L1</u>
B DOWN THEN UP	
A DOWN THEN UP	

QUESTIONS:

3. Draw the Exclusive - Or Logic using only NOR elements.

DESCRIPTION (EXPANDER):

When one input to a NAND gate is taken from the output of an AND gate the inputs of the AND gate become additional NAND inputs to the logic element. Thus a four input NAND gate can be expanded to a seven input NAND by using an AND gate.



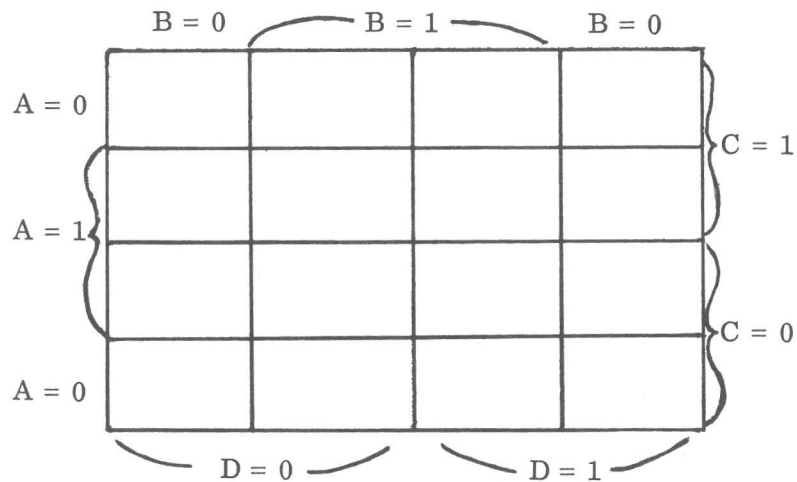
Experiment Logic 22-3
NAND EXPANDER

PROCEDURE:

Wire the logic. Test the system for NAND logic. The output will be a "0" only when all inputs are "1".

QUESTIONS:

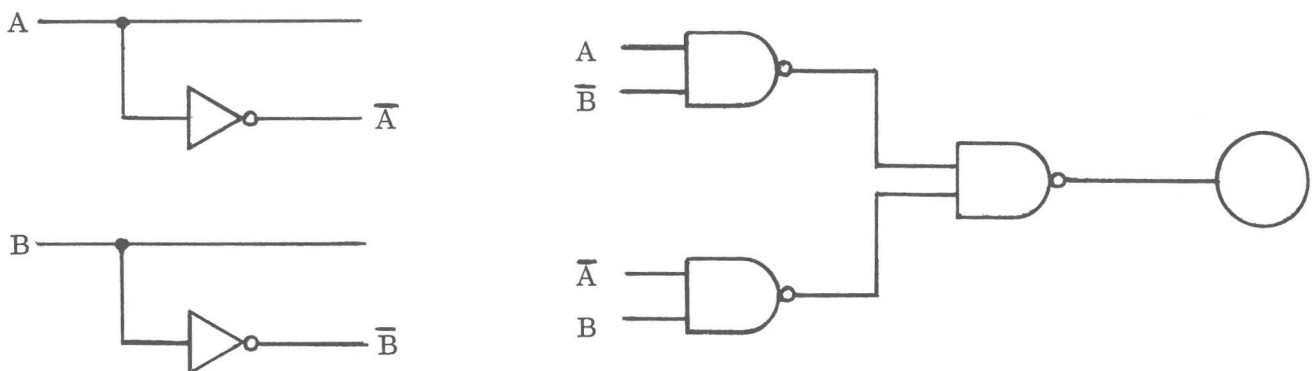
1. Draw the 4 input Karnaugh Map.



2. Draw the equivalent NOR system.

DESCRIPTION (EXCLUSIVE - OR):

Three NAND elements are connected as shown below to provide the Exclusive - Or function.

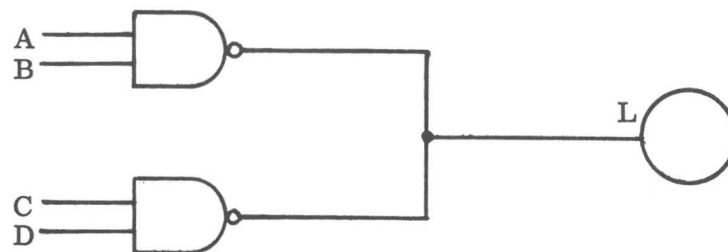


Experiment Logic 22-2

NAND EXCLUSIVE - OR

PROCEDURE:

Wire the logic shown below:



Experiment Logic 22-1
PARALLEL NANDS

Check the outputs. List various input combinations.

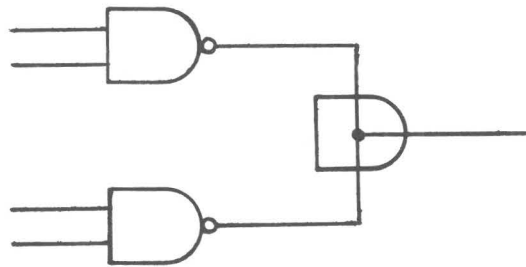
[illegible]

OBJECTIVE:

This experiment illustrates various NAND applications including NANDs connected in parallel, NANDs used in an Exclusive - Or configuration, and EXPANDER implementation.

DESCRIPTION (PARALLEL OUTPUT):

The outputs resulting when logic elements are connected in parallel depend on the particular electrical circuits used in the logic module. The NANDs in this system are "DTL" and have a single transistor (collector) output. When either NAND output is set to "O" or ground, then the output is ground or "O". The logic equation is then the AND of the outputs since a "1" is only present when both outputs are a "1". The parallel connection is sometimes drawn as shown below:



PARALLEL NOTATION

The logic equation is:

$$L = (\overline{AB}) (\overline{CD})$$

This can also be written as:

$$L = (\overline{A} + \overline{B}) (\overline{C} + \overline{D})$$

or it can be written as:

$$L = \overline{AB + CD}$$

PROCEDURE:

Wire the experiment for each diagram. Set up the input switches and complete the truth table for each logic function. Check the output to see that the appropriate logic functions are performed.

A	B	L AND	L NOT	L OR	✓
0	0				
0	1				
1	0				
1	1				

QUESTIONS:

3. Draw a NAND logic system for the function $L = A\bar{B}\bar{C}$.
4. Draw a NAND logic system for the function $L = A\bar{B} + \bar{C}$.

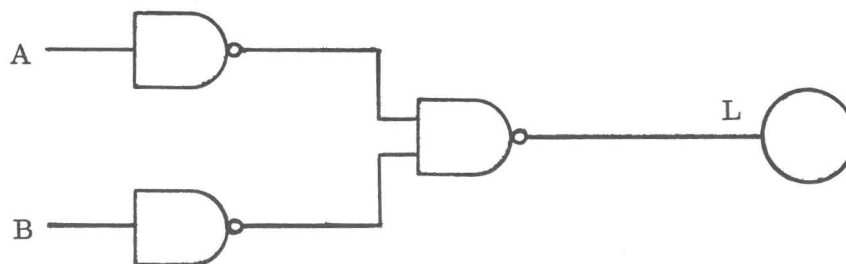
The NOT gate function is provided by using the NAND with only a single input as shown in Diagram 21 - 3.



Experiment Logic 21-3

NOT LOGIC

The OR gate is provided by inverting the inputs to the NAND gate as shown in Diagram 21 - 4.



Experiment Logic 21-4

OR LOGIC

PROCEDURE (NAND Element):

The TRUTH TABLE for the NAND circuit is below:

A	B	C	L (1)	L (2)	✓

First fill in the 8 conditions for A, B, C. Fill in column (1) with the expected NAND outputs. Then wire the experiment and enter the output data in column (2) and check both results.

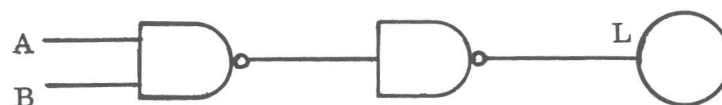
QUESTIONS:

1. Which input condition (A, B, C) is the only condition which results in a "0" output?

2. Draw the Logic MAP and VENN Diagram for the NAND circuit.

DESCRIPTION (NAND UNIVERSAL LOGIC):

The NAND output is an inverted AND. By inverting the output of the NAND again, the result is an AND as illustrated in DIAGRAM 21 - 1.



Experiment Logic 21-2

AND LOGIC

OBJECTIVE:

This experiment introduces the NAND Logic Circuit.

DESCRIPTION:

The equation for the NAND Logic Circuit is $L = \overline{ABC}$. It reads "L equals NOT A AND B AND C". The result is the same as an AND circuit followed by a NOT circuit as shown below:



AND - NOT

NAND is short for NOT-AND. The symbol is shown below and is the AND symbol with a circle (O) for inversion at the output.



Experiment Logic 21-1

NAND LOGIC

QUESTIONS:

1. Consider the following two questions:

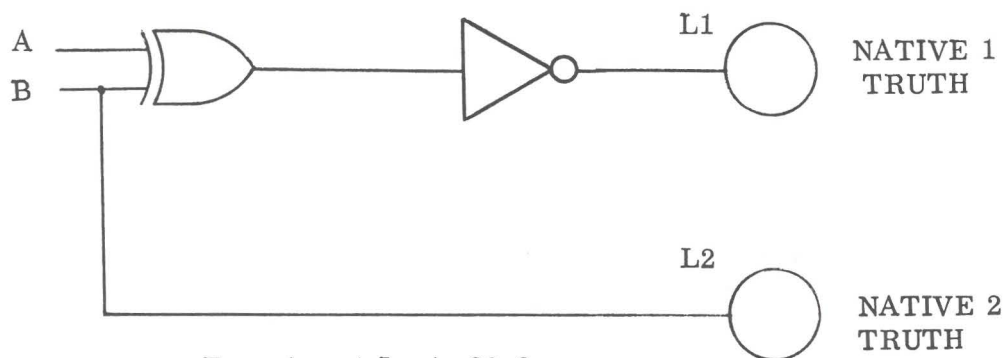
Question A (Native 1)	Are you both from the same tribe?
Question B (Native 2)	Are you both from the same tribe?

Will these questions work? Prepare a truth table and draw the logic diagram.

2. Can you think of another set of questions? Don't waste time on this.
3. Devise a simpler logic for the NATIVE 2 TRUTH condition.

EXCLUSIVE-OR SOLUTION

Note that NATIVE 2 is a TRUTH native when B=1. NATIVE 1 is a TRUTH native when both inputs A, B are the same or the NOT of the exclusive-or. The logic is:



Experiment Logic 20-2

EX-OR SOLUTION

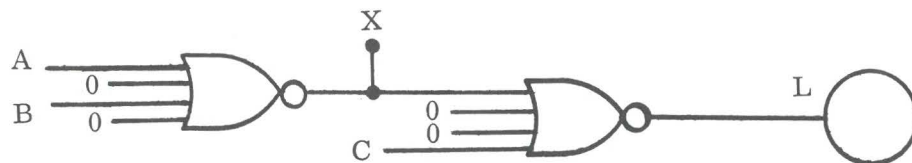
Wire the above logic in Figure 20-2. Check all combinations with the truth table. Is this logic correct? _____.

OBJECTIVE:

This experiment illustrates various standard logic configurations using NORs.

DESCRIPTION:

The diagram below uses 2 NORs with 3 inputs (A, B, C). An intermediate logic signal is X.



Experiment Logic 26 - 1

SERIES NOR

PROCEDURE(Serial NOR):

Write the equations or expressions for X and L.

X=

L=

Prepare the truth table with columns headed A, B, C, X, L(1) and L(2). Fill in the columns including expected X and L(1). Wire the experiment and enter the output in column L(2). Check the results.

A	B	C	X	L(1)	L(2)	✓

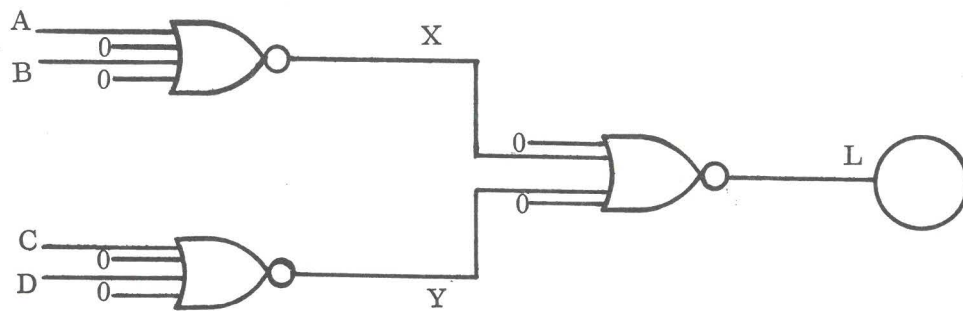
TABLE 26 - 1

QUESTIONS:

1. Draw an all NAND system for the same function.
2. Draw an AND, OR, NOT circuit for the same function.
3. Draw the VENN diagram for this system.

DESCRIPTION (OR - AND Configuration):

The logic in the diagram below performs the function derived below:



Experiment Logic 26 - 2

OR - AND

The output X is given by the equation for the NOR and is:

$$X = \overline{A+B}$$

Similarly, Y is given by:

$$Y = \overline{C+D}$$

The output L is given by:

$$L = \overline{X+Y}$$

$$L = \overline{\overline{A+B} + \overline{C+D}}$$

Using the DeMorgan's Theorem, this becomes:

$$L = (A+B) \cdot (C+D)$$

PROCEDURE:

Wire the logic.

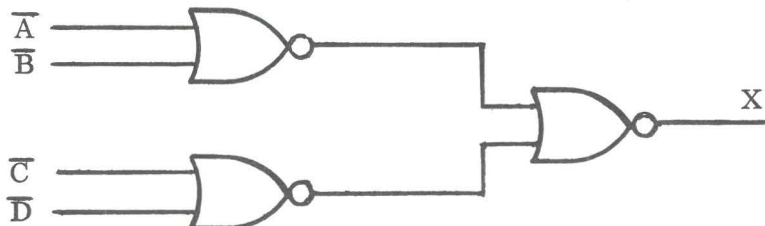
A	B	C	D	L(1)	L(2)	✓

TABLE 26 - 2

Select various combinations of inputs and enter them in the data table. Determine the expected output L and enter this in column L(1). Set the inputs and record the logic output. Enter the data in column L(2).

QUESTIONS:

4. Draw the all NAND and the OR-AND logic. Compare both configurations.
5. What is the Logic given by:



QUESTIONS:

1. Draw the Karnough Map.
2. Draw a logic diagram using AND, OR, NOT.
3. Repeat using only NORs.
4. Repeat using only NANDs.

OBJECTIVE:

This technical note presents the rules and relationships used in Boolean Algebra, with illustrative examples and experiments.

DESCRIPTION:

1. The logical operators are:

AND = $\cdot$ (dot) or (nothing between symbols)

OR = $+$ (plus sign)

NOT = $\bar{}$ (bar over the letter)

2. Some logic equations are:

$L = AB$ (L equals A and B) (also $L = A \cdot B$)

$L = A + B$ (L equals A or B)

$L = \bar{A}$ (L equals NOT A)

$L = \bar{A} + B$ (L equals NOT A or B)

$L = \overline{AB}$ (L equals NOT(A AND B) = NAND)

$L = \overline{A + B}$ (L equals NOT(A OR B) = NOR)

$L = A \oplus B$ L equals EXCLUSIVE-OR for A, B.

3. The rules or techniques for manipulating equations are similar to those in algebra:

(1) Commutative: i. e. $A + B = B + A$

$$AB = BA$$

(2) Associative: i. e. $A + B + C = A + (B + C) = (A + B) + C$

$$ABC = A(BC) = (AB)C$$

(3) Distributive: i. e. $A(B + C) = AB + AC$

$$(A + B)(A + C) = AA + AC + AB + BC$$

$$= A + BC$$

4. Useful identities are:

- (1) $\overline{\overline{A}} = A$
- (2) $A \cdot 1 = A$
- (3) $A \cdot 0 = 0$
- (4) $A \cdot A = A$
- (5) $A \cdot A \cdot A = A$
- (6) $A \overline{A} = 0$
- (7) $A + 1 = 1$
- (8) $A + 0 = A$
- (9) $A + A = A$
- (10) $A + \overline{A} = 1$

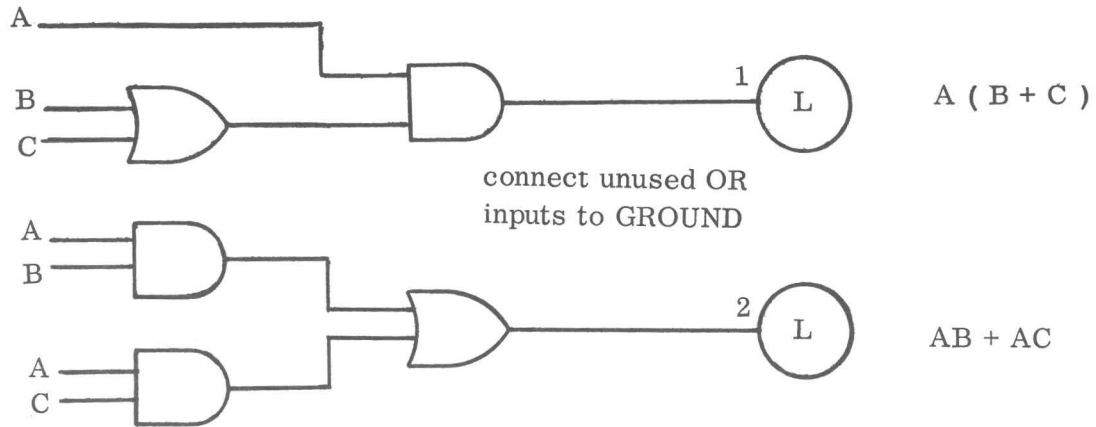
5. Other useful "theorems" are:

- (1) $A + AB = A$
- (2) $A (A+B) = A$
- (3) $(A+B) (A+C) = A + BC$
- (4) $A + \overline{A}B = A + B$
- (5) $A (\overline{A} + B) = A B$
- (6) $(A + B) (\overline{A} + C) = AC + \overline{A}B$
- (7) $AB + \overline{A}C = (A + C) (\overline{A} + B)$
- (8) $A B C = \overline{\overline{A} + \overline{B} + \overline{C}}$ (DeMorgans Theorem)

PROCEDURE:

- (1) Select any 5 rules or theorems from above and verify them by

implementing the logic of both sides of the equation and by comparing the outputs for all input variations . An example is shown below for the DISTRIBUTIVE LAW.



Experiment Logic 28 - 1
DISTRIBUTIVE LAW

Enter the data in the TRUTH TABLE below:

A	B	C	L(1)	L(2)	✓

TABLE 28 - 1

(2) Repeat for four additional rules.

(3) A TAUTOLOGY is a logic statement which is always true and correct for all values of variables.

Consider the logic equation:

$$A B + A \bar{B} + \bar{A} \bar{B} = A + \bar{B}$$

Is this Logic STATEMENT or EQUATION a TAUTOLOGY?

Prepare the Truth Table for both sides of the equation:

$$X = A B + A \bar{B} + \bar{A} \bar{B}$$

$$Y = A + \bar{B}$$

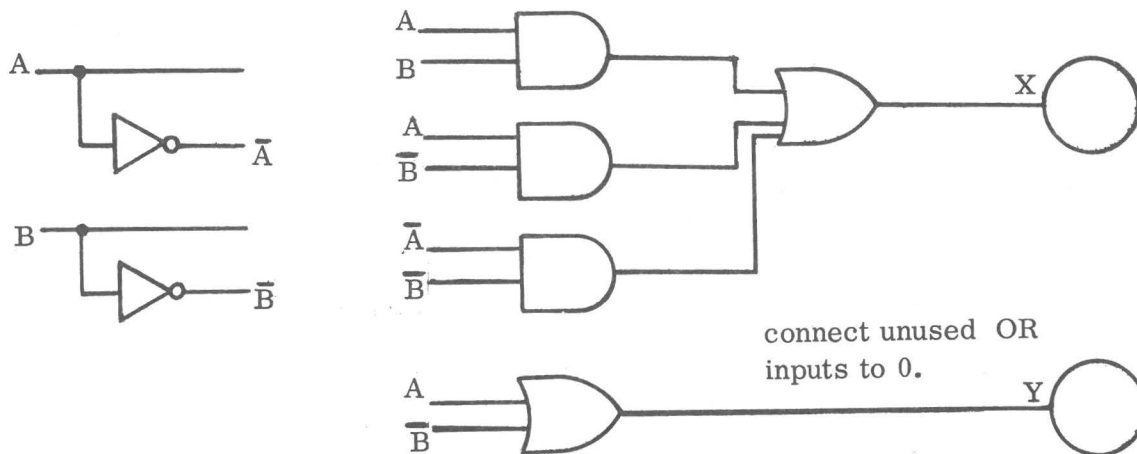
A	B	X	Y
0	0		
0	1		
1	0		
1	1		

TABLE 28 - 2

They are identical and the statement is a TAUTOLOGY.

PROCEDURE(Tautology):

Wire the Logic shown on the following page to verify the above Tautology.



Experiment Logic 28 - 2

TAUTOLOGY

Enter the data in the table below :

A	B	X	Y

TABLE 28 - 3

QUESTIONS:

1. Is $\bar{A} B = A + \bar{B}$ a TAUTOLOGY ?
2. Is $\bar{A} + B + \bar{A} \bar{B}$ a TAUTOLOGY?
3. Is $(A + B) (A + C) = A + B C$ a TAUTOLOGY ?
4. Is $A + \bar{A} B = A + B$ a TAUTOLOGY ?

OBJECTIVE:

This experiment describes NEGATIVE LOGIC and the DUALITY of logic systems, and De-Morgan's Theorem.

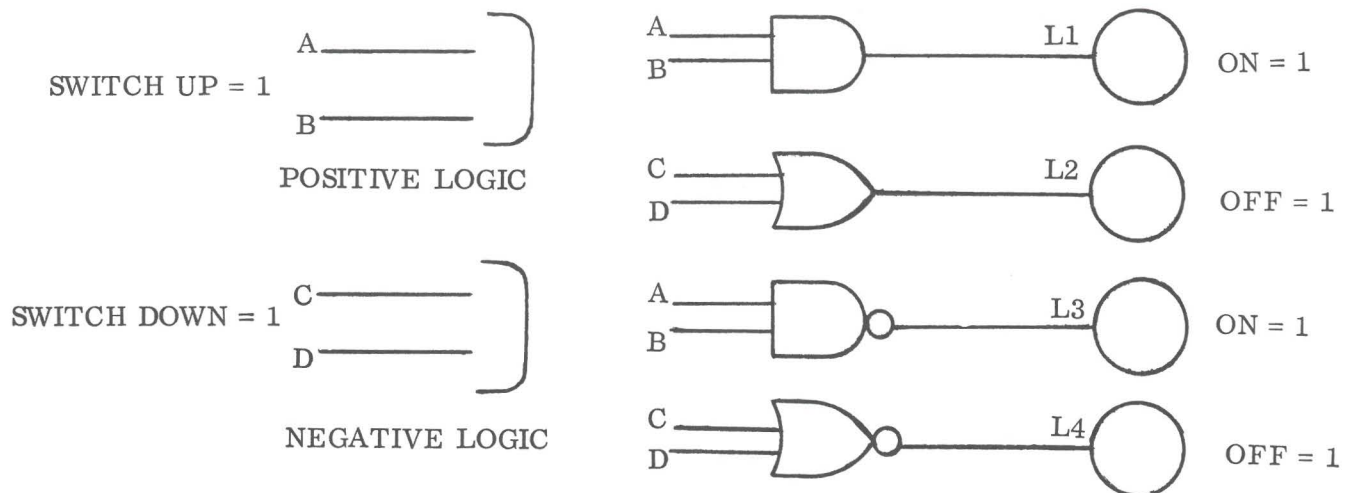
DESCRIPTION(Negative Logic):

In all prior experiments, the UP switch position and the ON lamp condition represented a "1", and the "1" was given by the + voltage. The "0" was given by the negative "ground" voltage or 0 volts. This notation is called POSITIVE LOGIC because the "1" is the more positive voltage. It is also possible for the "1" to be represented by the more negative voltage. This notation is called NEGATIVE logic. DUALITY refers to the fact that the fact that the NEGATIVE LOGIC "OR" generates the same output as the POSITIVE LOGIC "AND". The NEGATIVE LOGIC "NOR" produces the same output as the POSITIVE LOGIC "NAND".

PROCEDURE:

Switch A and B inputs are used with POSITIVE LOGIC (i.e. UP=1). Switch C and D inputs are used with NEGATIVE LOGIC (i.e. DOWN=1). Lamps L1 and L3 are used with POSITIVE LOGIC NOTATIONS (i.e. ON= 1). Lamp L2 and L4 are used with NEGATIVE LOGIC NOTATIONS (i.e. OFF= 1).

Set all conditions of A, B, C, D. Read all 4 outputs and enter data in Table. Verify the DUALITY of NEGATIVE LOGIC.



* ALL UNUSED OR & NOR
INPUTS TO BE RETURNED
TO "0".

Experiment Logic 29 - 1

NEGATIVE LOGIC

(All future experiments are positive logic.)

QUESTIONS:

1. Why is negative logic used?
2. The following equation is used in a negative logic system.

$$X = AB + \overline{C}$$

Write the same equation for positive logic.

3. Draw the LOGIC Diagram for the above in both a Negative Logic and Positive Logic form.

DESCRIPTION (DeMorgan's Theorem):

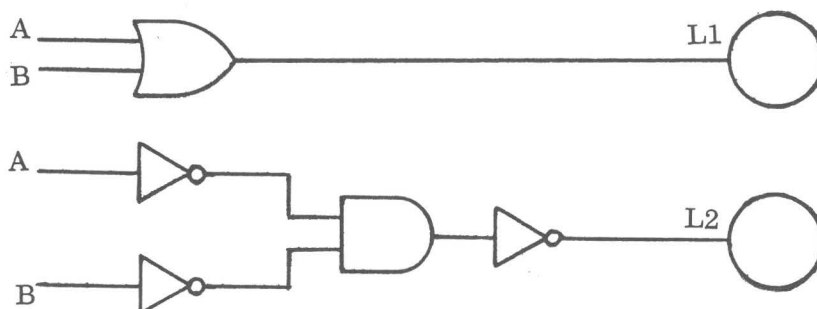
DeMorgan's Theorem for manipulating logic equations is the following:

$$X = A + B = \overline{\overline{A} \cdot \overline{B}} \quad \text{or} \quad Y = A \cdot B = \overline{\overline{A} + \overline{B}}$$

It states that where two (or more) terms are related by a logical AND or OR, then one can change that relationship (i. e. AND to OR and OR to AND) by changing the term to the NOT term (i. e. A to $\overline{A}$) and by then changing the entire result to its NOT.

PROCEDURE:

Wire the experiment below which has the two equivalent outputs. Check both outputs and enter them in the DATA table.



QUESTIONS:

4. Using DeMorgan's Theorem, simplify

$$L = \overline{(\overline{A} + \overline{BC})}.$$

5. Draw the logic for $L = \overline{A} + \overline{B} + C$ using NANDs. Use DeMorgan's Theorem on the above equation and draw the simplified logic.

6. Use DeMorgan's Theorem on $L = \overline{(A + \overline{B}) \cdot (\overline{C} + D)}$ to yield

$$L = \overline{A} B + C \overline{D}$$



Ed-Lab

EXPERIMENT MANUAL

ARITHMETIC TECHNIQUES and SEQUENTIAL SYSTEMS

UNIT NO.

2

CES INDUSTRIES, INC.

EXPERIMENT MANUAL

Copyright © 1974, 1976, 1978, 1980, 1981 CES INDUSTRIES, INC.

Name _____

Course _____

OBJECTIVE:

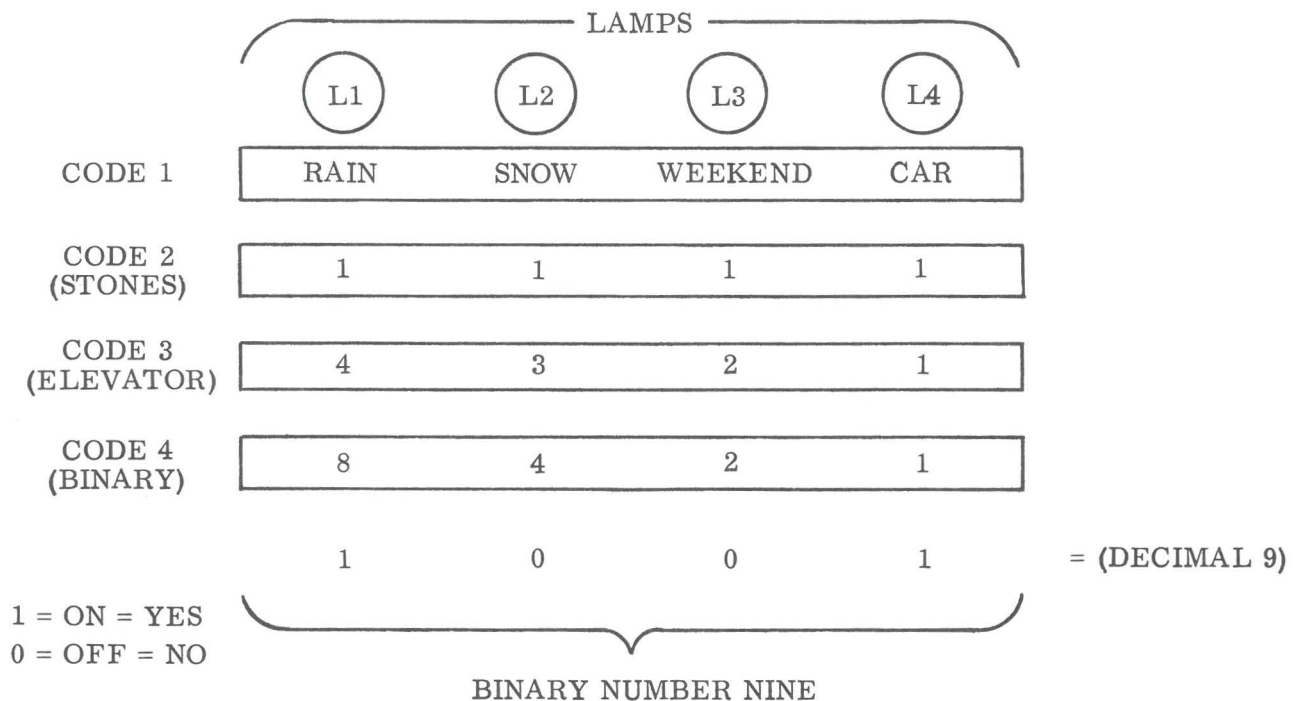
The purpose of this experiment is to illustrate the coding of the binary number system.

DESCRIPTION:

The switches were used in earlier experiments to signify whether or not it was RAINING, COLD, WEEKEND, etc. The assignment of a meaning or code to each signal line is "coding" the line. Code 1 below is for the items referred to, and any number of switches can be "1" at the same time. With A, C, and D up, it is raining on the weekend and we have a car. Code 2 has a number "1" for each switch line. If 3 switches are up, the number is 3.

Code 3 has a different number for each line covering 1-4. When switch C is up, lamp 3 is ON and the number is 3, etc. This code is used when displaying the floor at which an ELEVATOR is on. The line numbered 3 is not needed; it can be set as A and B, i. e., 1 plus 2.

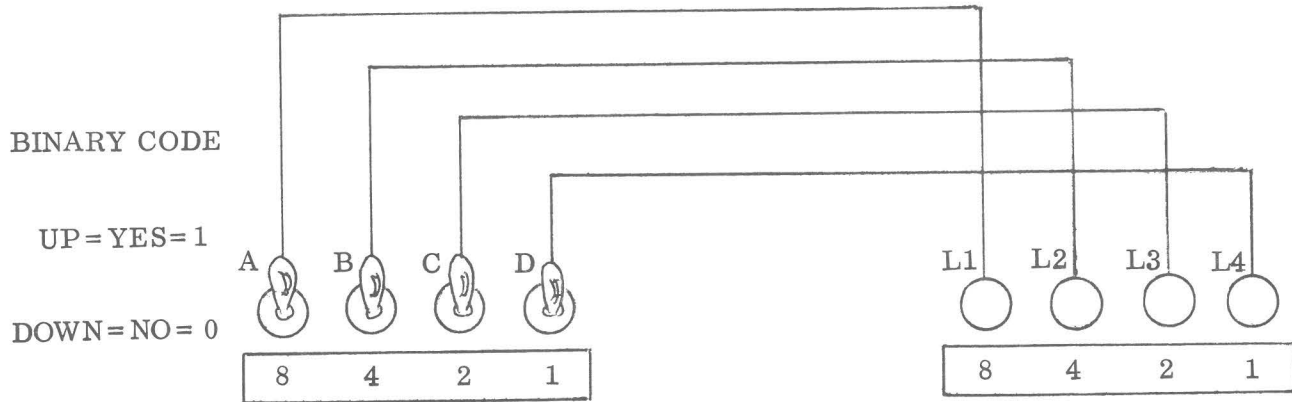
Code 4 is the binary code and is 8, 4, 2, 1. Numbers 0, 1, to 15 can be coded. Thus number 9 is the 8 and 1 lamps ON.



For numbers greater than 4, we can combine two switches , and this FLOOR 5 can be given by lamps 4 and 1 both ON. We also do not need 6 and 7 which can be combinations of 1, 2, 4.

PROCEDURE:

Set the switches as shown for the binary numbers in the table. Fill in the missing codes. Check each number.



Experiment Logic 30 - 1 BINARY CODING

No.	A (8)	B (4)	C (2)	D (1)
0	0	0	0	0
3	0	0	1	1
5	0	1	0	1
12				
15				
7				
8				

QUESTIONS:

1. What is the binary code for 14?

3. What would the binary code be for 27? (Use more switch positions.)

DESCRIPTION (Binary and Octal Coding):

Binary bits only have the digits 1 and 0. The number of binary bits becomes large for large numbers. Octal notation is used to simplify binary notation.

The maximum number of different numbers (including 0) for a number of binary bits is given in the table below:

binary bits (N)	different decimal numbers = 2^N
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1024
11	2048
12	4096

The binary number 001 011010 is actually the decimal number 90 (i.e. $64 + 16 + 8 + 2$). The octal number system uses the binary bits in groups of 3 with the "decimal number" notation from 0 thru 7 (which is then the octal number notation). Thus:

$\underbrace{001}_1 \underbrace{011}_3 \underbrace{010}_2$ — BINARY
 — OCTAL

the octal number is 132. This is consistent with the binary system and provides a shorthand notation. A subscript after a letter denotes the "base" or "radix" (i.e. decimal (10) or octal (8) or binary (2) to which the number is written). Thus X equal to ninety is :

$$X_2 = 001011010 \quad (\text{IN BINARY})$$

$$X_8 = 132 \quad (\text{IN OCTAL})$$

$$X_{10} = 90 \quad (\text{IN DECIMAL})$$

and all are equivalent.

EXERCISES:

4. Write these numbers to all bases (2, 8, 10).

$$X_2 = 010110$$

$$X_8 =$$

$$X_{10} =$$

$$Y_8 = 147$$

$$Y_2 =$$

$$Y_{10} =$$

$$Z_{10} = 2391$$

$$Z_8 =$$

$$Z_2 =$$

DESCRIPTION (NUMERICAL - BINARY CONVERTER):

The TRUTH TABLE below serves as the guide for encoding from numerical to binary code.

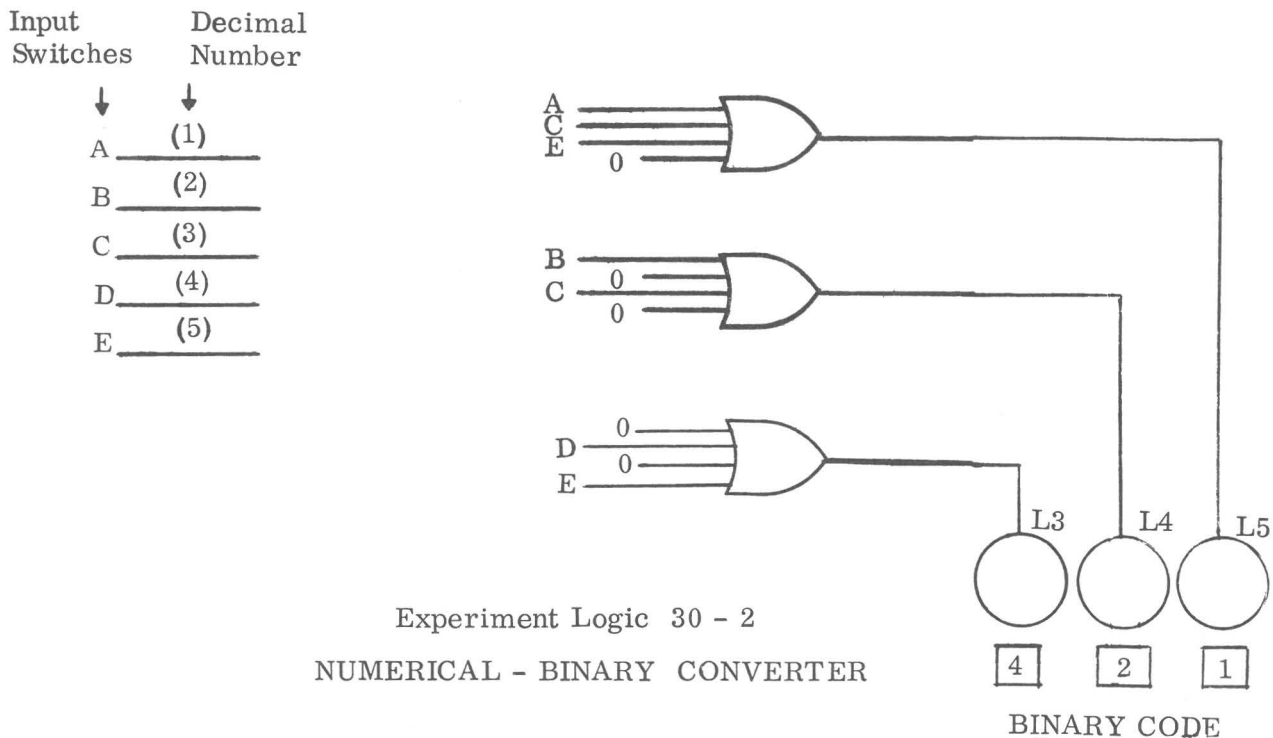
DATA:

Decimal	(4)	(2)	(1)	✓
0				
1				
2				
3				
4				
5				
6				
7				

The binary bit (1) is used in coding all odd numericals (i. e. 1, 3, 5, 7, etc.). The second binary bit (2) is part of the numerical digits 2, 3, 6, 7 etc. The third binary bit (4) is used in the numericals 4, 5, 6, 7 etc. The OR gates shown in Fig 30 - 2 perform the coding function to convert the decimal to a binary number.

PROCEDURE:

Wire the experiment logic shown below :



Set the input switches to the proper decimal codes. Check the binary outputs (lamps).

QUESTIONS:

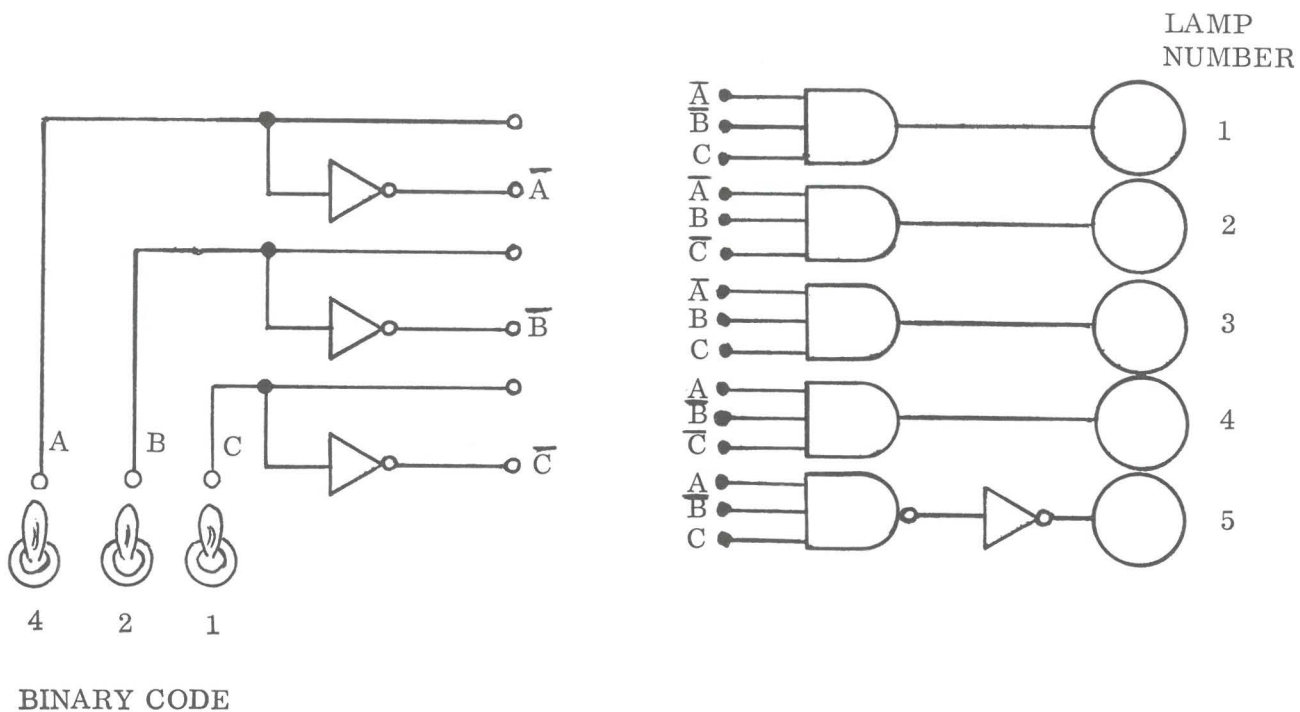
5. What are the full OR gate inputs to convert the decimals 0 to 9 into binary numbers?

OBJECTIVE:

Computers operate using the binary code while the operators prefer the conventional recognizable decimal numbers. The logic in this experiment converts binary numbers to decimal outputs.

DESCRIPTION (Binary to Numerical Converter) :

The input switches, ABC, provide a binary coded input with a 4,2,1 code. The corresponding decimal numbers are displayed on the output lamps. A 3-input AND gate decodes the binary numbers and drives the output display lamp. Only the proper combination will result in an output to the proper lamp.



Experiment Logic 31-1

NUMERAL CONVERTER

PROCEDURE:

Wire the logic shown above. Set the switches to the binary numbers shown in the truth table. Record the outputs in the last column. (Rewire 2 gates to provide outputs for 6 and 7.)

BINARY			DECIMAL	✓
A (4)	B (2)	C (1)	NUMERAL	
0	0	0	0	
0	0	1	1	
0	1	0	2	
0	1	1	3	
1	0	0	4	
1	0	1	5	
1	1	0	6	
1	1	1	7	

QUESTIONS:

1. If 4 binary inputs (coded 8, 4, 2, 1) were used, what gate inputs (i. e. 8 or $\overline{8}$, 4 or $\overline{4}$, etc.) would be used in decoding the decimal number 13?

DESCRIPTION (Binary to Decimal Conversion):

This technical note describes techniques for converting a binary number to its decimal equivalent.

METHOD 1 :

Consider the Binary Number 1 1 0 1 0. Each bit position corresponds to a decimal number, i.e.

16	8	4	2	1	decimal
1	1	0	1	0	binary

Add all decimal equivalents together where the binary number is a "1" .

$$1 \times 16 = 16$$

$$1 \times 8 = 8$$

$$0 \times 4 = 0$$

$$1 \times 2 = 2$$

$$0 \times 1 = \underline{1}$$

$$\text{TOTAL DECIMAL} = 26$$

METHOD 2:

Consider the same Binary Number 1 1 0 1 0 . Take the most significant digit (left) . Go to the next digit and double the partial sum (1 from the most significant digit) and add 1 if this next digit is a 1. Continue to all digits and the last sum is the decimal number.

		Binary number		Partial Sum
binary number		1	→ 1 =	1
		1	double = 2 plus 1 =	3
		0	double = 6 plus 0 =	6
		1	double = 12 plus 1 =	13
		0	double = 26 plus 0 =	<u>26</u>

Decimal answer is 26.

EXERCISES:

Convert the following binary numbers to their decimal equivalents
(use both methods).

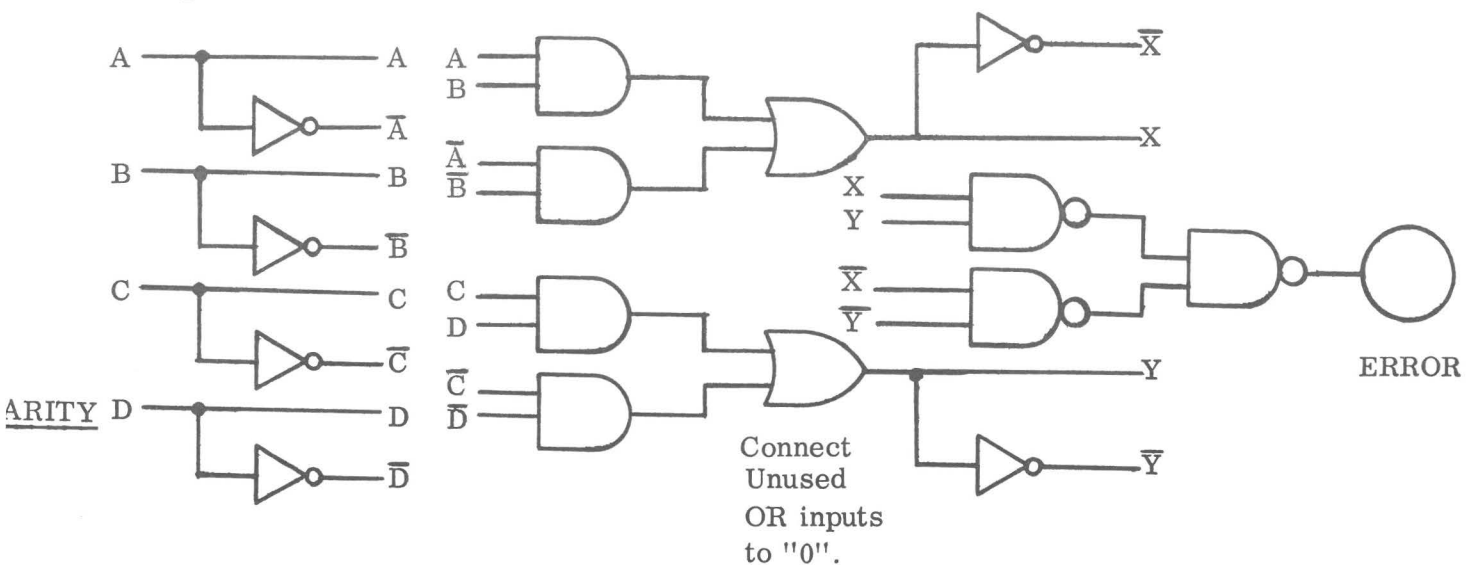
2. 1 0 1 1 0
3. 1 0 1 0 1
4. 0 1 1 1 0 1
5. 0 1 0 1 1 0 1 1 0 1

OBJECTIVE:

To describe PARITY and the function of PARITY to detect errors.
An ODD PARITY ERROR detector and EVEN PARITY ENCODER are implemented.

DESCRIPTION (ODD Parity Error):

The three binary digits can be used to encode eight items or numbers 0 thru 7. If there is an error in setting or transmitting any single bit, then the result is another number which is an error. A parity bit is an additional bit which enables the detection of any single error. When this PARITY BIT is set to result in an odd number of 1's, it is called ODD parity. When set to an even number of 1's, it is called EVEN parity. This experiment uses ODD parity. The TRUTH TABLE is given on the next page. The switches set the number (A, B, C) plus parity (D). An error in switch settings results in an error light to be ON.



Experiment Logic 32 - 1
ODD PARTY ERROR DETECTOR

PROCEDURE:

Wire the logic. Set the input switches and check the operation of the parity error lamp by setting in a wrong parity.

A	B	C	D	PARITY ERROR
1	0	1	0	
1	0	1	1	

TABLE 32 - 1

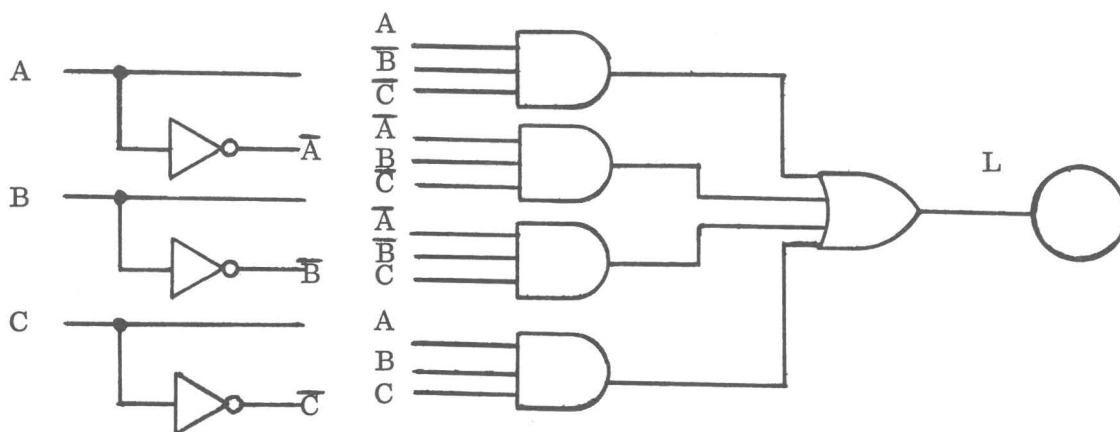
EXERCISE:

- Using 3 inputs (A, B, C) draw the logic to provide an EVEN parity bit, (L).

DESCRIPTION (Even Parity Error):

The Even Parity Error Logic generates an output "1" for an ODD number of input "1"s (A, B, C). The result is therefore an even number of 1's. This is an EVEN parity generator.

The logic shown consists of the logic terms of each set (A, B, C) in the truth table.



Experiment Logic 32 - 2
EVEN PARITY ENCODER

$$L = \overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} + \overline{A}B\overline{C} + ABC$$

PROCEDURE:

Wire the Logic and check the output in the truth table.

A	B	C	L	✓
0	0	0		
0,	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

QUESTIONS:

2. Draw the logic using only NAND gates.
3. Draw the Karnaugh Map.
4. How many terms would there be in the equation for 4 inputs?
5. Draw the Logic using EXCLUSIVE - OR elements.

OBJECTIVE:

This experiment summarizes various additional binary codes. A CODE CONVERTER is also implemented.

DESCRIPTION:

The decimal numbers 0 thru 9 are normally encoded using four binary bits coded as 8, 4, 2, 1, in what is referred to as a B C D (Binary Coded Decimal) Code. The Binary number 53, is then

$\overbrace{0101}^5 \overbrace{0011}^3$	DECIMAL
0 1 0 1 0 0 1 1	B C D

Because 4 bits yield numbers 0 to 15 and only 0-9 are needed, alternate weighted codes can be used:

A 7421 B C D Code would encode the number 8 (decimal) as 1 0 0 1 (binary). Other codes include:

2421
5421
6311
7421
5311
5211
4221

QUESTIONS:

1. Select any 3 of the above codes, and encode the decimal numbers 57 and 369.

EXCESS THREE CODE:

The EXCESS THREE code encodes a number in BINARY as 3 above the decimal number. Thus, 5 (decimal) would be encoded as 8 or 1000 (binary). This is advantageous because the maximum binary number (15) is 6 higher than the maximum decimal number (9). When adding two numbers in Excess three, the SUMS require no correction when there is an overflow.

NEGATIVE WEIGHTED CODES:

A negative weighted code is:

(16) (-8) (4) (-2) (1)

The number 5 is encoded as:

0 0 1 0 1

The number 3 is encoded as:

0 0 1 1 1

QUESTIONS:

2. Encode all decimal numbers (0 to 9) and (0 to -9) in the above code.

FIVE BIT CODES:

Various 5 bit codes have advantages. The 2 OUT OF 5 code is advantageous in communications where an error is easily detectable.

The 2-out-of-5 code is:

Decimal	2-out-of-5
0	0 0 0 1 1
1	0 0 1 0 1
2	0 0 1 1 0
3	0 1 0 0 1
4	0 1 0 1 0
5	0 1 1 0 0
6	1 0 0 0 1
7	1 0 1 0 0
8	1 0 1 0 0
9	1 1 0 0 0

PROCEDURE:

The TRUTH TABLE for the binary B C D and 2421 codes is given in the diagram on the following page. The logic is given in the diagram on the following page.

Wire the experiment and check the conversion.

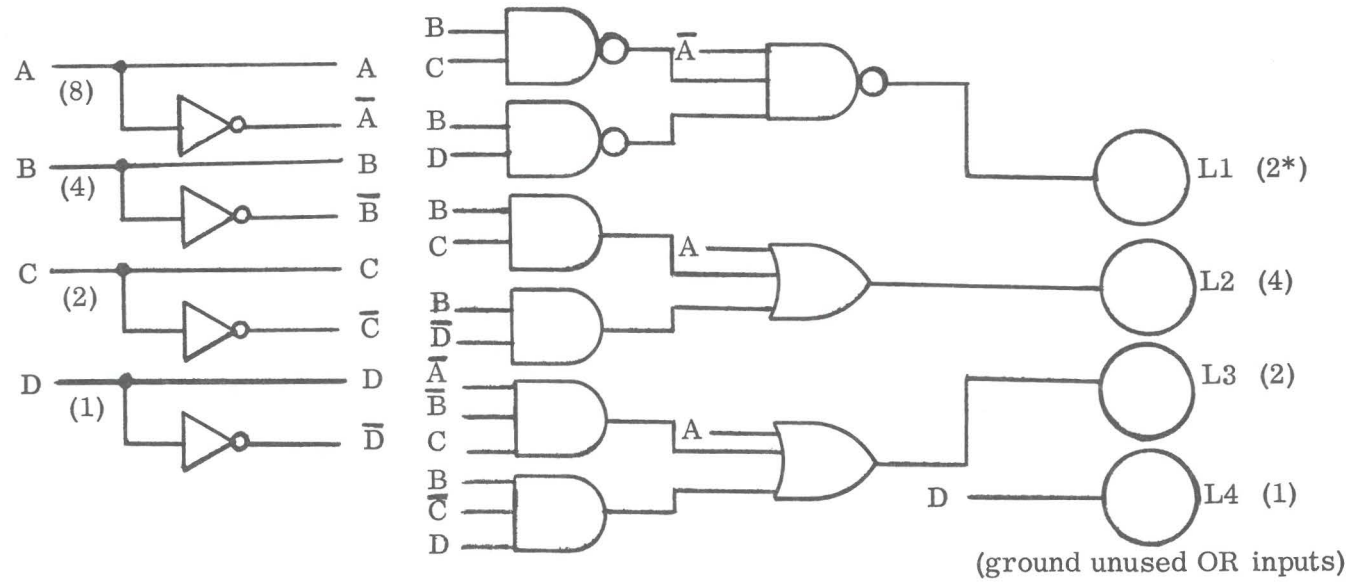
QUESTIONS:

3. Draw the logic for a 2*421 to 8421 converter.

(2* is used to show the difference between the
2* and 2 bits)

TRUTH TABLE BCD / 2421

A	B	C	D	#	L1	L2	L3	L4
(8)	(4)	(2)	(1)	Decimal	2*	4	2	1
0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	1
0	0	1	0	2	0	0	1	0
0	0	1	1	3	0	0	1	1
0	1	0	0	4	0	1	0	0
0	1	0	1	5	1	0	1	1
0	1	1	0	6	1	1	0	0
0	1	1	1	7	1	1	0	1
1	0	0	0	8	1	1	1	0
1	0	0	1	9	1	1	1	1



Experiment Logic 33-1
BCD TO 2421 CONVERTER

OBJECTIVE:

The purpose of this experiment is to illustrate the usefulness of the GRAY CODE or Reflected Binary Code. A Gray to Binary Converter is implemented.

DESCRIPTION:

In the normal binary code, more than one bit may change from one number to another as in $3 = 011$ and $4 = 100$ where all 3 bits change. Because these changes may not all be perfectly "synchronized", other numbers appear in-between. This can cause errors in some systems. The Gray Code is shown in the table below:

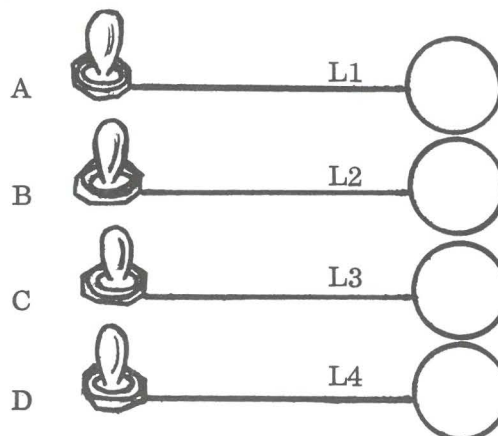
Only one bit changes between each number.

← GRAY →				#	Binary			
A	B	C	D		A	B	C	D
0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	1
0	0	1	1	2	0	0	1	0
0	0	1	0	3	0	0	1	1
0	1	1	0	4	0	1	0	0
0	1	1	1	5	0	1	0	1
0	1	0	1	6	0	1	1	0
0	1	0	0	7	0	1	1	1
1	1	0	0	8	1	0	0	0
1	1	0	1	9	1	0	0	1
1	1	1	1	10	1	0	1	0
1	1	1	0	11	1	0	1	1
1	0	1	0	12	1	1	0	0
1	0	1	1	13	1	1	0	1
1	0	0	1	14	1	1	1	0
1	0	0	0	15	1	1	1	1

PROCEDURE (Gray Code):

Wire the experiment.

Experiment Logic 34-1
GRAY CODE



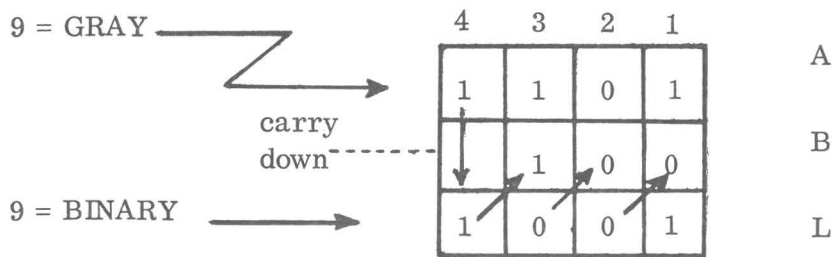
Use ABCD as a binary number set to 0000. Using only one finger, set the switches from decimal 0 thru 7. Observe that from 1 to 2 (001 to 010) we must go thru zero (001 to 000 to 010) or thru three (001 to 011 to 010). Use ABCD for the GRAY CODE and observe how this does not occur.

QUESTIONS:

1. Thru which numbers can the binary number change when switching from 3 (011) and 4 (100)?

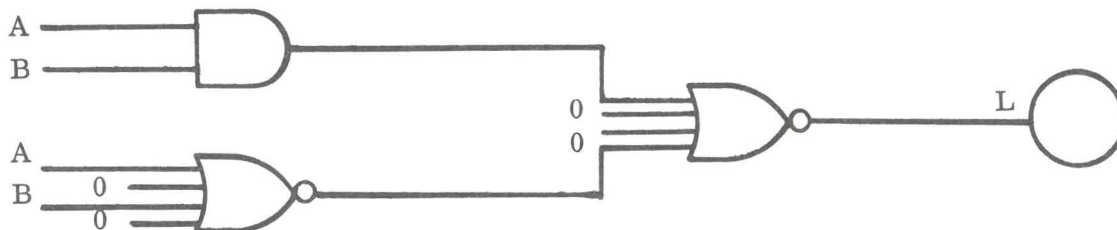
DESCRIPTION (Gray to Binary):

The basic logic uses the EXCLUSIVE -OR to generate the binary code. The output of the EXCLUSIVE - OR is the binary bit to be determined. The inputs are: A = the gray code bit; B = the previous most significant binary bit. Example: The gray code is 1101 or 9. The bits are numbered 4,3,2,1. The first binary coded bit (left side) is the same as the gray bit or "1". The next binary bit uses this "1" (B) and the gray bit (A,3) "1" to generate the EXCLUSIVE - OR binary code output of "0". See table below:



PROCEDURE (Gray to Binary):

Wire the experiment Exclusive-OR, on the following page. Set Switch B to the previous binary bit (start with 0). The lamp reads the next binary bit. Try various examples in the tables on the following page:



Experiment Logic 34-1
 GRAY TO BINARY CONVERTER

GRAY	A	0	1	1	1	#5
	B					
BINARY	L	0				
GRAY	A	1	0	1	0	#12
	B					
BINARY	L					
GRAY	A					
	B					
BINARY	L					

Binary to Gray:

A similar Procedure is used in converting from Binary to Gray:

		4	3	2	1	
9 = BINARY	→	1	0	0	1	A
	↘	0	1	0	0	B
9 = GRAY	→	1	1	0	1	L

This time, however, the input B is the previous Binary number.
Using the same logic, convert various Binary Numbers to the

Gray equivalent:

BINARY	A				
	B	0			
GRAY	L				
BINARY	A				
	B	0			
GRAY	L				
BINARY	A				
	B				
GRAY	L				

QUESTIONS:

- Write the GRAY CODES for the decimals 124, 125 and 126 .

OBJECTIVE:

This note summarizes the rules used in the addition of binary numbers and demonstrates the operation of the HALF-ADDER.

DESCRIPTION (Addition):

The addition of 2 binary digits is summarized below. The sum of 2 bits is referred to as HALF ADDITION and performed by a logic HALF ADDER.

A	0	0	1	1
B	<u>+0</u>	<u>+1</u>	<u>+0</u>	<u>+1</u>
sum	0	1	1 (carry 1	0 sum)

The sum of 3 binary digits is required when adding 2 numbers plus a previous carry and is summarized below. The logic for this addition is called a FULL ADDER.

A	B	previous carry	sum	new carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	1	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

TABLE 35 - 1

EXERCISES:

Complete the following Examples:

				(1)					(2)
5	0 1 0 1			3	0 0 1 1			2	0 0 1 0
+ 4	<u>0 1 0 0</u>			+ 7	<u>0 1 1 1</u>			+11	<u>1 0 1 1</u>
	1 0 0 0	carry		—	1 1 0				—
9	1 0 0 1	sum			1 0				—

3. Prepare a Table for Subtraction to generate a DIFFERENCE and a BORROW.

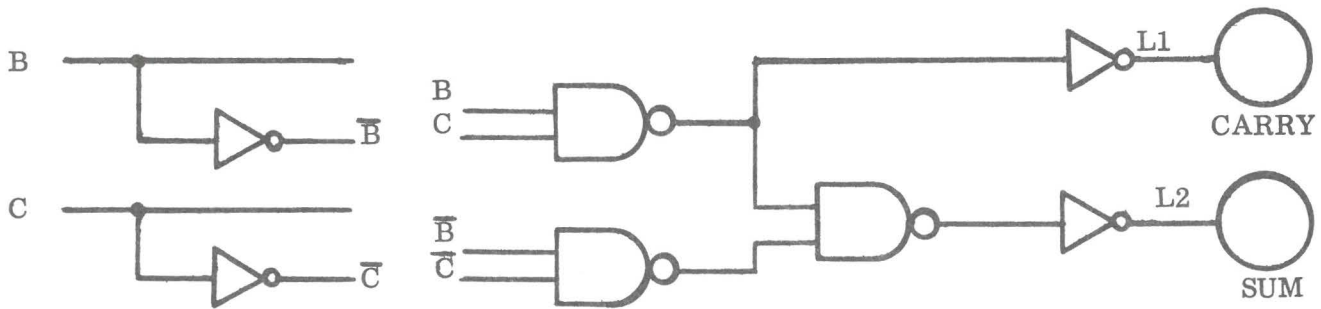
Complete the examples:

4.		5	0 1 0 1		5.	7	0 1 1 1
		- 4	<u>0 1 0 0</u>			- 3	<u>0 0 1 1</u>
			_____ borrow				_____ borrow
			_____ difference				_____ difference

This experiment demonstrates how the Half Adder is implemented using only NAND elements.

DESCRIPTION (NAND Half Adder):

The Half Adder logic accepts two inputs (B and C in this experiment) and provides two outputs; a SUM and a CARRY. The above logic performs this function. Inverters are used to provide the NOT function of the input although NAND elements could have also been used.



Experiment Logic 35 - 1
NAND HALF ADDER

PROCEDURE:

Wire the logic as given above. Vary the inputs and check the outputs. Enter the data in the table.

TABLE 35 - 2

B	C	sum	carry
0	0		
0	1		
1	0		
1	1		

QUESTIONS:

- Draw the logic for an all NOR Half - Adder.
- Draw the logic for an AND - OR - NOT Half Adder.

8. Draw the Logic for a HALF-ADDER using the EXCLUSIVE-OR plus any other elements.
9. Compare the relative complexity of each.

OBJECTIVE:

This experiment demonstrates a FULL ADDER and how it is used to perform addition of 2 binary numbers by adding each bit "consecutively" or "serially".

DESCRIPTION:

The FULL ADDER truth table is given below.

OLD			SUM NEW		
A	B	C	S	K	
0	0	0	0	0	
0	0	1	1	0	
0	1	0	1	0	
0	1	1	0	1	
1	0	0	1	0	
1	0	1	0	1	
1	1	0	0	1	
1	1	1	1	1	

The logic equations are

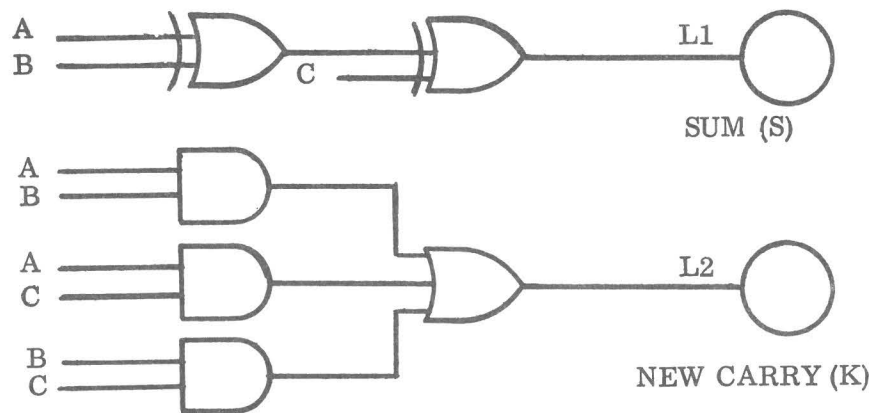
$$\text{SUM} = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$$

$$\text{CARRY} = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC = AB + AC + BC.$$

PROCEDURE:

Wire the logic on the following page. Verify the TRUTH TABLE with all combinations. Perform serial addition of 2 numbers in the following manner:

	8	4	2	1	Decimal
A	0	1	1	1	7
B	0	0	1	1	+3
Carry C			y	0	
Sum				x	10



Experiment Logic 36 - 1
FULL ADDER

Set the carry (Switch C) to 0 (DOWN) and A and B as in column (1) ($A=1$, $B=1$). Read the SUM (L1) and enter the bit in box X. Read the CARRY (L2) and enter the bit in box Y. Repeat the process for column (2) (then (4), (8)). Set A, B, and C, etc. Repeat adding ($2 + 11$) and ($5 + 4$).

QUESTIONS:

1. What happens if we try to add ($12 + 9$) in the above system using 4 bits?
2. How can we detect this overflow?
3. What is the limit to the number of bits that can be added using the above adder.
4. Draw a full adder with only NANDs.

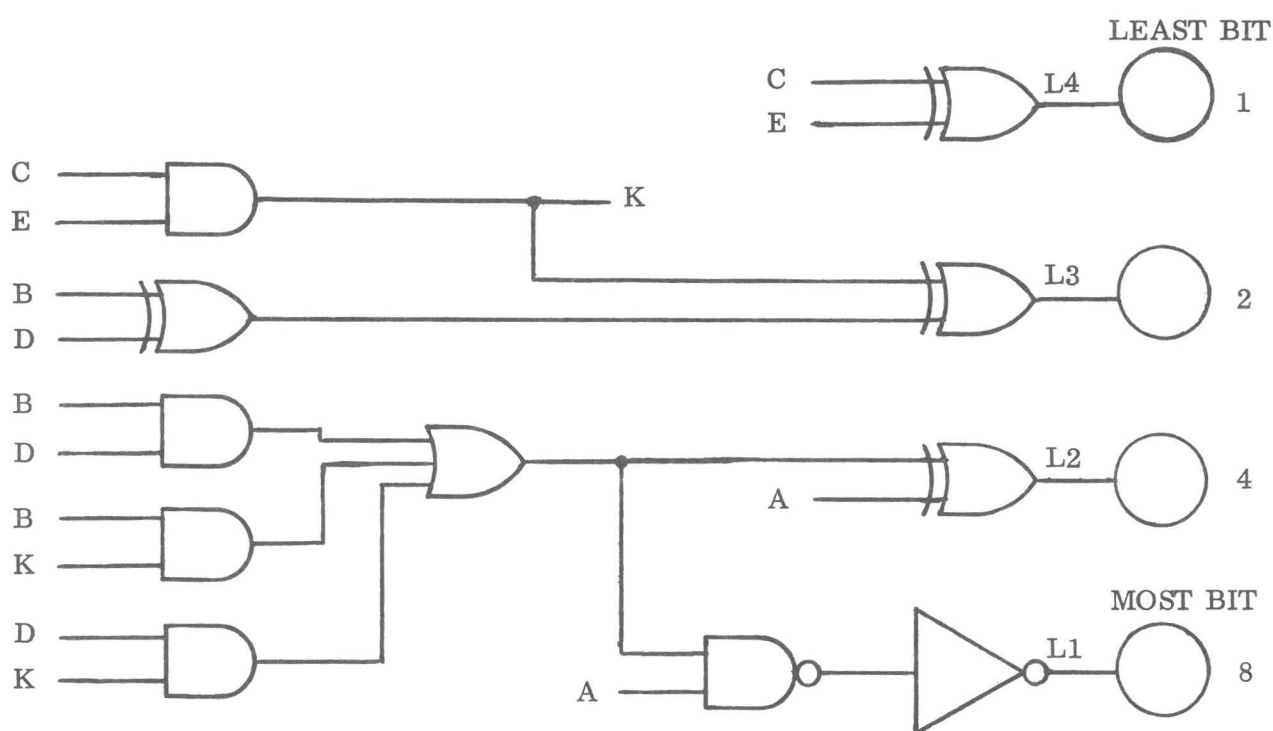
OBJECTIVE:

This experiment demonstrates a parallel binary adder for two binary numbers.

$$\begin{array}{r} \text{ABC} \\ + \text{DE} \\ \hline = \end{array}$$

DESCRIPTION:

The previous experiment processed or added each bit in a "serial" operation. Thus a 4 bit number would take 4 steps; one bit at a time. A parallel adder adds all bits together in one step but, as expected, would then require more logic elements. The above experiment uses a Half-Adder for the first output (there is no previous carry) and the Full Adder for the next bit. Number ABC has a maximum of 7, and DE has a maximum of 3.



Experiment Logic 37 -1

PARALLEL ADDER

PROCEDURE:

The data sheet on the following page shows each input number (ABC and DE) in binary and decimal form and the total sum (L1, L2, L3, L4) in

binary and decimal form. Set the input switches and read the outputs and verify the operation. Select other inputs to complete the table.

A	B	C	#1	D	E	#2	L1	L2	L3	L4	SUM
0	1	0	2	1	1	3	0	1	0	1	5
1	1	0	6	1	0	2					

QUESTIONS:

1. What is the total number of different input combinations ?
2. How many "adders" are needed for a parallel 5 bit addition machine?

OBJECTIVE:

Thus far, only Positive Binary Numbers have been used. This experiment introduces the "complement notations" for negative numbers and illustrates how it is used in adding negative numbers and for subtraction.

DESCRIPTION (Complements):

A complement of a bit is its NOT or inverse. The complement of 1 is 0 and of 0 is 1, and this is called the "ones" complement. An extra higher order bit is added to a number to specify the sign. Consider a 4 bit number 0 1 0 1 which is (+5). An extra 0 is added for the sign (0 0 1 0 1). The complement is then written (1 1 0 1 0). This represents minus 5 and is the ONE'S COMPLEMENT.

When 1 is added to the number the result is the TWO'S COMPLEMENT (1 1 0 1 0 + 0 0 0 0 1 = 1 1 0 1 1). The TWO'S COMPLEMENT (1 1 0 1 1) also represents (-5) minus 5. The first (left side) bit denotes the sign. A "1" indicates a negative number. Addition with the TWO'S COMPLEMENT is performed in the regular manner. Add +9 to -5.

	+9	0 1 0 0 1
two's complement	-5	1 1 0 1 1
carry		1 1 0 1 1 0
sum	+4	0 0 1 0 0

The overflow carry is ignored. The answer 0 0 1 0 0 is +4. Add -7 to +5. Plus 7 is (0 0 1 1 1). The 1's complement is (1 1 0 0 0).

	-7	1 1 0 0 1
	+5	0 0 1 0 1
		0 0 0 1 0
sum		1 1 1 1 0

The answer is negative. The 1's complement is 0 0 0 0 1. The 2's complement (add 1) is 0 0 0 1 0 or (-2). The answer is (-2). Numbers are subtracted by using the 2's complement. To obtain the difference (23 - 6), add 23 to the 2's complement of 6.

QUESTIONS:

1. Add: 8 plus -3 in binary.
2. Perform the subtraction : (8-3).

3. Add: +13 to +7 using a total of 5 bits and 2's complement notation. (Note: If there is a 1 in the first (left) bit of the answer without a 1 in either of the two numbers, then there is an overflow.) (If there is a 0 in the left bit of the sum without a 0 in either of the numbers, there is an overflow.)

4. Subtract: +5 minus -4.

DESCRIPTION (Fractions):

This table summarizes the characteristics of the binary numbers for n bits. The 2^n column indicates the number of different numbers which can be represented. The 2^{-n} column represents the value of the least significant bit for fractional number.

n	2^n	2^{-n}
0	1	1.0
1	2	0.5
2	4	0.25
3	8	0.125
4	16	0.0625
5	32	0.03125
6	64	0.015625
7	128	0.0078125
8	256	0.00390625
9	512	0.001953125
10	1,024	0.0009765625
11	2,048	0.00048828125
12	4,096	0.000244140625
13	8,192	0.0001220703125
14	16,384	0.00006103515625
15	32,768	0.000030517578125

DESCRIPTION (Table):

1. A typical ten bit number is 0 1 1 0 1 0 1 1 0 1. The least significant bit is valued at "1". The lowest number is 0. The highest number is 1023. The number of different numbers is 1024 (including 0). The most significant

bit is valued at 512.

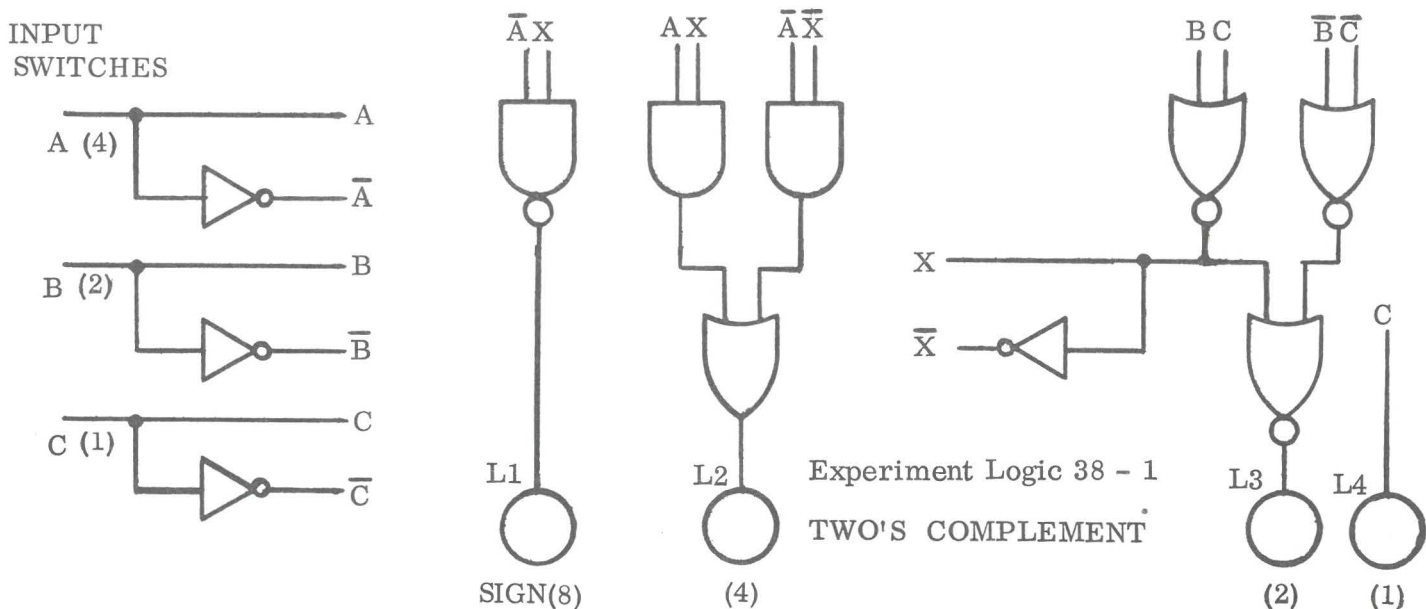
2. A typical four bit fractional number is .1001. The most significant bit (left) is valued at 0.5. The least significant bit is valued at 0.0625 .

QUESTIONS:

5. What are the decimal equivalent for the two above binary numbers?
6. Write the binary equivalent for 28.
7. Write the decimal equivalent for 1001.
8. Write the decimal equivalent for .101
9. Write the decimal equivalent for 101.011
10. Write the "closest" binary equivalent for 22.6 using 5 bits to the left of the decimal point and 4 fractional bits to the right of the decimal point.

DESCRIPTION (Experiment):

The 2's complement is generated by reversing all bits and adding a 1. This requires a half-adder per bit. The first bit (right) is the same because reversing the bit and adding 1 is the same digit. If the first (right) input is a 0, then there must be a carry. The second stage is a Half-Adder. The next stage is a carry (inverted) because it is added to an extra 1 bit.



PROCEDURE:

Wire the logic. Set the input switches to all numbers ABC and enter the 2's complement in the table.

A	B	C	L1	L2	L3	L4
0	0	0	0	0	0	0
0	0	1	1	1	1	1
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

QUESTIONS:

7. Draw the logic diagram to generate the 1's complement.
8. Draw the logic diagram to generate the 9's complement of the 2*421 code (Note: 3 is the 9's complement of 6 and 4 is the 9's complement of 5.) Comment on the result.
9. The following numbers are in 2's complement form. What are their decimal equivalents?

0011
1000

OBJECTIVE:

This experiment describes and demonstrates Binary Multiplication.

RULES:

The following binary multiplication rules are similar to those used in decimal multiplication for (A times B equals C).

$$\begin{aligned} A \times B &= C \\ 0 \times 0 &= 0 \\ 1 \times 0 &= 0 \\ 0 \times 1 &= 0 \\ 1 \times 1 &= 1 \end{aligned}$$

MULTIPLICATION:

Binary multiplication is performed in the same manner as decimal multiplication by summing the partial products.

Consider $6 \times 5 = ?$

$\begin{array}{r} 6 \\ \times 5 \\ \hline \end{array}$	$\begin{array}{r} 0110 \\ \times 0101 \\ \hline 0110 \\ 0000 \\ 0110 \\ 0000 \\ \hline 0011110 \end{array}$	multiplicand multiplier add partial products sum of partial products
--	---	--

The binary answer is equivalent to the decimal number 30 which is the correct answer.

EXERCISES:

1. Multiply 1 0 0 1 and 1 1 0 1 and check the answer.
2. Multiply 1 1 1 1 and 1 1 1 1 and check the answer.
3. What is the maximum number of binary digits in the product resulting from the multiplication of two 4 bit numbers?

4. Multiply the following fractional binary numbers 1 0 . 1 0 and 1 1 . 0 1 and check the answer.

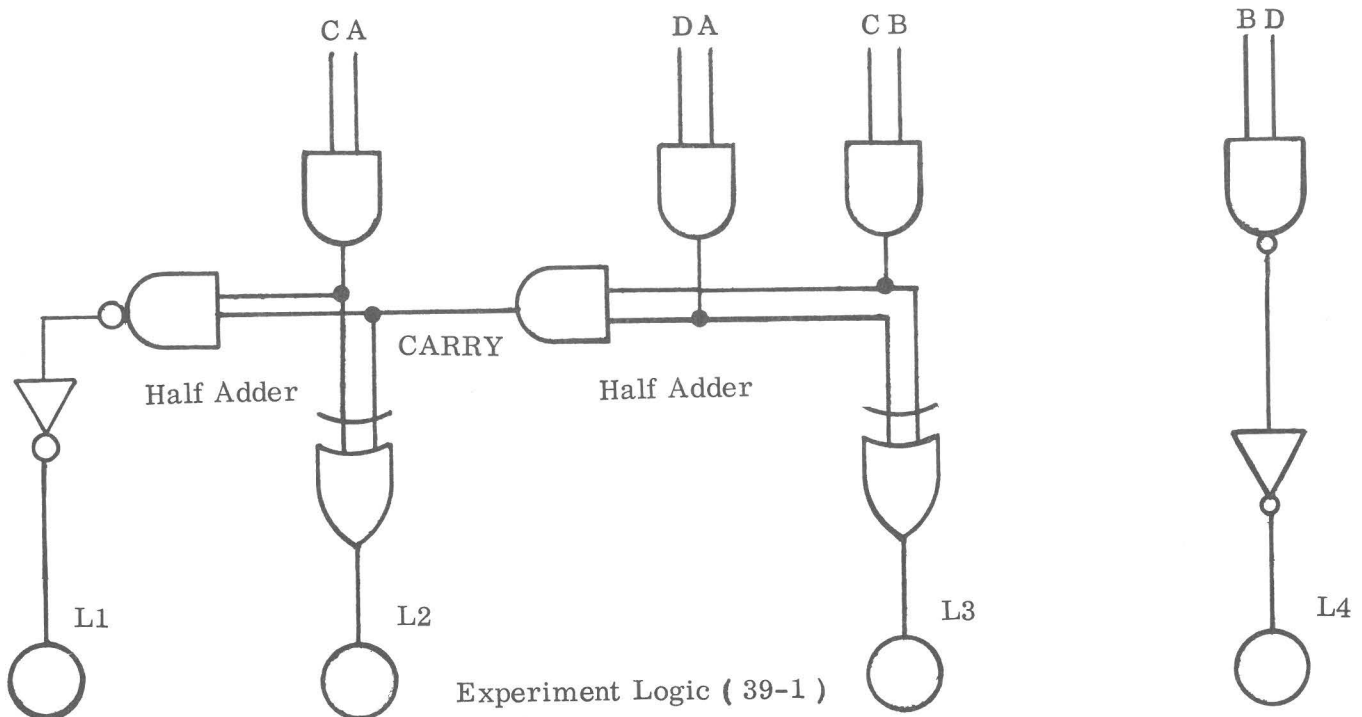
5. Multiply .1 0 1 1 and .0 0 1 1 and round-off the answer to the first four bits to the right of the decimal point. What is the maximum round-off error?

DESCRIPTION:

This experiment demonstrates a binary parallel multiplier for 2 bit numbers. Consider two binary numbers with the bits denoted as AB and CD. There are four possible combinations for each number. (0 0 , 0 1 , 1 0 , 1 1 , i.e. decimal 0 to 3) and the product output ranges from 0 to 9 requiring 4 output bits. The multiplication is implemented with the addition of partial products. With AB as the multiplier and CD as the multiplicand, the product is given as the sum of the partial products below:

$$\begin{array}{r} \text{CD} \quad \text{set to 00 if B = 0} \\ + \text{CD} \quad \text{set to 00 if A = 0} \\ \hline \text{sum} \end{array}$$

The logic diagram is given below and includes two half-adders.



Experiment Logic (39-1)

MULTIPLICATION

PROCEDURE:

Wire the logic. Select numbers for AB and CD and check the results.

A	B	# 1	C	D	# 2	L1	L2	L3	L4	

QUESTIONS:

6. How many different input combinations are possible?
7. How many different output combinations are possible?

OBJECTIVE:

This experiment describes other radix number systems and demonstrates a 3 stable state or TERNARY memory.

DESCRIPTION:

Decimal numbers have ten digits or numerals (0, 1, 2, etc. to 9). Binary numbers have two digits (0, 1). Consider a three numeral system:

Decimal	Ternary
0	00
1	01
2	02
3	10
4	11
5	12
6	20
7	21
8	22

QUESTIONS:

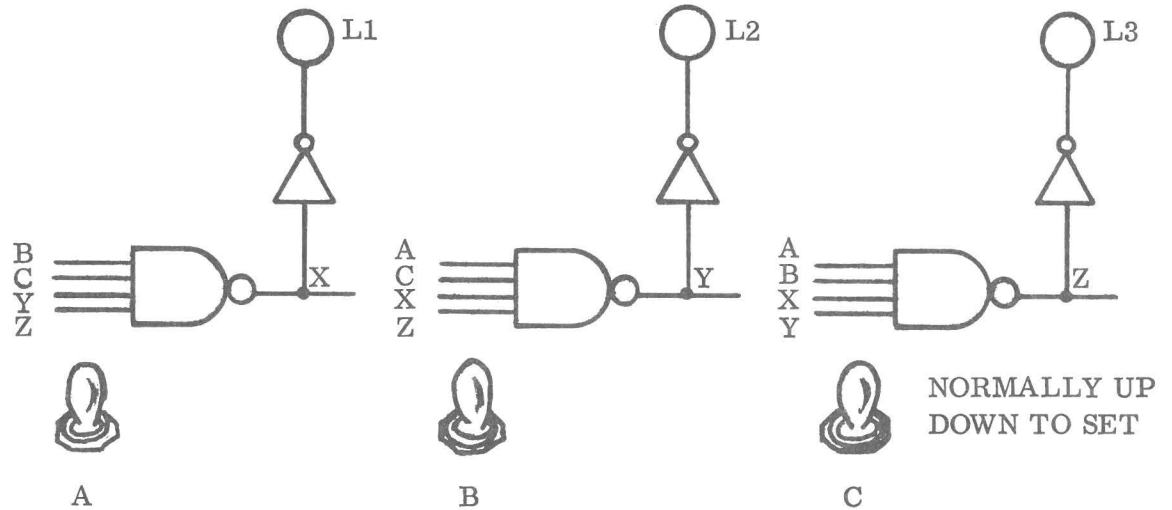
1. Make a table with decimal numbers 0 to 20 with columns for a "four radix" system, an "eight radix system" (octal), a "16 radix system" (hexadecimal), (Use letter symbols A, B, etc., for numerals larger than 9).

Radix nomenclatures are:

Radix	System
2	Binary
3	Ternary
4	Quaternary
5	Quinary
6	Hexary
8	Octal
9	Nonal
10	Decimal
12	Duodecimal
16	Hexadecimal

DESCRIPTION:

Logic elements are used to implement a 3 stable state memory. Three NAND elements are used as shown below.



Experiment Logic 40 - 1 - TERNARY LOGIC

The 3 switch inputs A, B, C are normally UP. When input A is switched DOWN, its signal input into the NANDs forces L2 and L3 to "0" which force L1 to "1". When input A is switched UP, there are only "1" inputs to the first NAND and its output remains in the "0" state with L1 at a "1". The same is true for the other switch inputs (B and C) which control lamps L2 and L3.

INPUT	OUTPUT
START ALL UP	
A DOWN - UP	
B DOWN - UP	
C DOWN - UP	
B DOWN - UP	

PROCEDURE:

Wire the logic. Set all the switches UP. Set the switches as shown in the table and record the outputs. Observe that the circuit "remembers" the last switch that was "DOWN".

QUESTIONS:

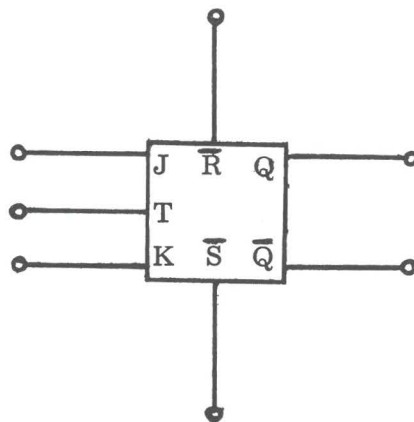
2. Design the logic for a 3 state memory using only NORs.

OBJECTIVE:

This experiment reviews the Flip-Flop Memory element symbol and demonstrates the Direct Set and Reset (Clear) Operation.

DESCRIPTION (Flip-Flop):

The flip-flop logic element has 5 inputs and 2 outputs, and power. The power connections are made internal to the trainer so only 7 points are available on the front panel. Two of the T points are available to limit tie point use. The inputs and outputs are denoted by the various letters shown below. These inputs and letters can be written on any side of the square and can be either inside or outside of the square for the convenience of the user.



FLIP - FLOP

SYMBOLS: Q AND $\bar{Q}$

OUTPUTS

These are the flip-flop complimentary outputs. When one is a "1" the other is a "0".

 $\bar{S}$ AND $\bar{R}$

DIRECT SET
AND CLEAR
INPUTS

These inputs are used to set and clear the flip-flop. A "0" at the $\bar{S}$ (direct set) input sets the flip-flop to $Q=1$. A "0" at the $\bar{R}$ (direct reset or direct clear) input clears the flip-flop to $Q=0$.

A flip-flop with these inputs is also called an "R-S" flip-flop.

J AND K STEERED
INPUTS

These inputs provide control signals to set and clear the flip-flop. They are either "1" or "0" but do not actually change the state of the flip-flop. The change is initiated by a Trigger.

T TRIGGER

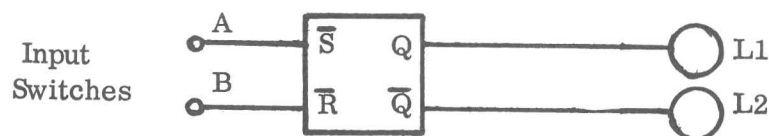
This input signal sets and clears the flip-flop in accordance with the J and K inputs.

DESCRIPTION(Direct Set and Clear):

The FLIP-FLOP memory element has two outputs Q and $\bar{Q}$. One is the Complement of the other. $\bar{S}$ IS THE DIRECT SET INPUT, $\bar{R}$ IS THE DIRECT CLEAR INPUT. Application of a "0" to the $\bar{S}$ input will cause the flip-flop to assume the $\bar{Q}=0$, $Q=1$ state. After the flip-flop has been set to any one state, the input which determined that state may be removed and the flip-flop will remain in its original state. This is the "memory" feature.

PROCEDURE:

Wire the logic shown:



Experiment Logic 41-1

SET - CLEAR FLIP - FLOP

Place both switches in the UP position. Set the input switches down then up one at a time. Record the conditions of the output lamps. Observe that the flip-flop "remembers" the last input condition. Apply zeros to both the direct set and the direct clear at the same time, and observe the results.

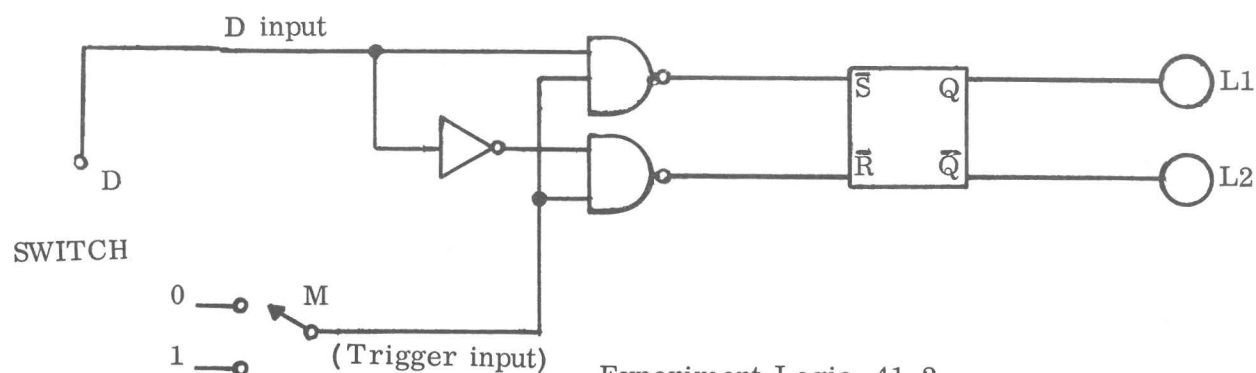
INPUT	L1	L2
A down - up		
B down - up		
B down - up		
A down - up		
A & B down		
A & B down - up		

QUESTIONS:

1. When $Q=1$, what is the value of $\bar{Q}$?
2. Connect the Q and $\bar{Q}$ output of one flip-flop to the $\bar{S}$ and $\bar{R}$ inputs of another and then switch the first one manually using switches A and B. What occurs?

DESCRIPTION (Single Input or Type D):

A single input is provided by a switch input D. When switch M is depressed, the input AND gates are energized and the flip-flop is set or cleared in accordance with the input. The input "0" state is used to perform the clear operation by means of an input Inverter.



Experiment Logic 41-2

TYPE D FLIP - FLOP

PROCEDURE:

Wire the logic. Set switch D at various input conditions and observe the output condition both before and after depressing the control switch M.

TEST	D INPUT	L1	L2

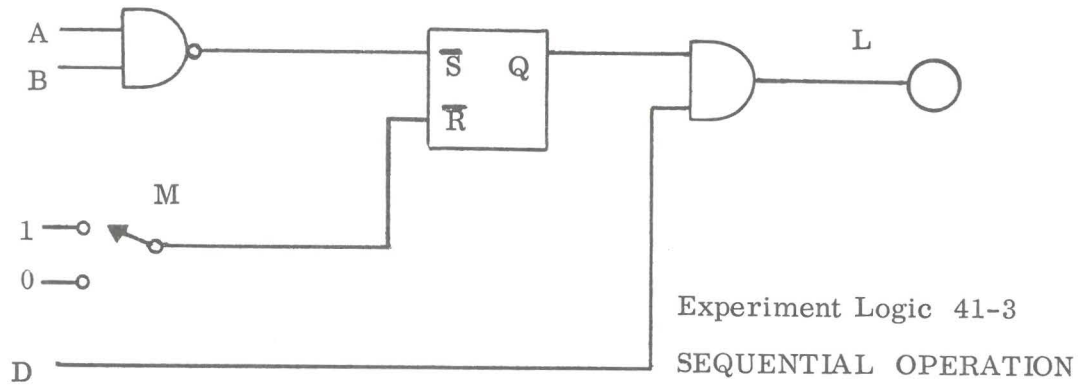
QUESTIONS:

3. Draw the logic for a system where a "0" input clears the flip-flop and a "1" input sets the flip-flop.

4. Draw the logic for a modified type D flip-flop where two inputs control the condition. Inputs A AND B clear the flip-flop, when switch M is depressed. This is equivalent to a TYPE D flip-flop with B as the input, M as the TRIGGER, and A as an energising control signal.

DESCRIPTION (Sequential Operation):

Switch M CLEARS the flip-flop (see figure below).



Switches A and B operate a NAND gate to SET the flip-flop. At any later time, switch D provides an input to another AND gate. The logic equation is $L=ABD$, but AB and D do not have to occur at the same time. Instead, they can occur in "sequence" where the flip-flop provides the memory.

PROCEDURE:

Wire the logic shown above. First set switches A and B down and depress switch M momentarily to clear the memory. Then set A and B (0 or 1) in accordance with the table. Then return A and B to the down position. Finally set switch D and record the output L. Check the results.

A	B	D	L
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

QUESTIONS:

5. Draw the logic to generate $L = AB + AC$ when A and B appear together first, are then reset, and later C appears.

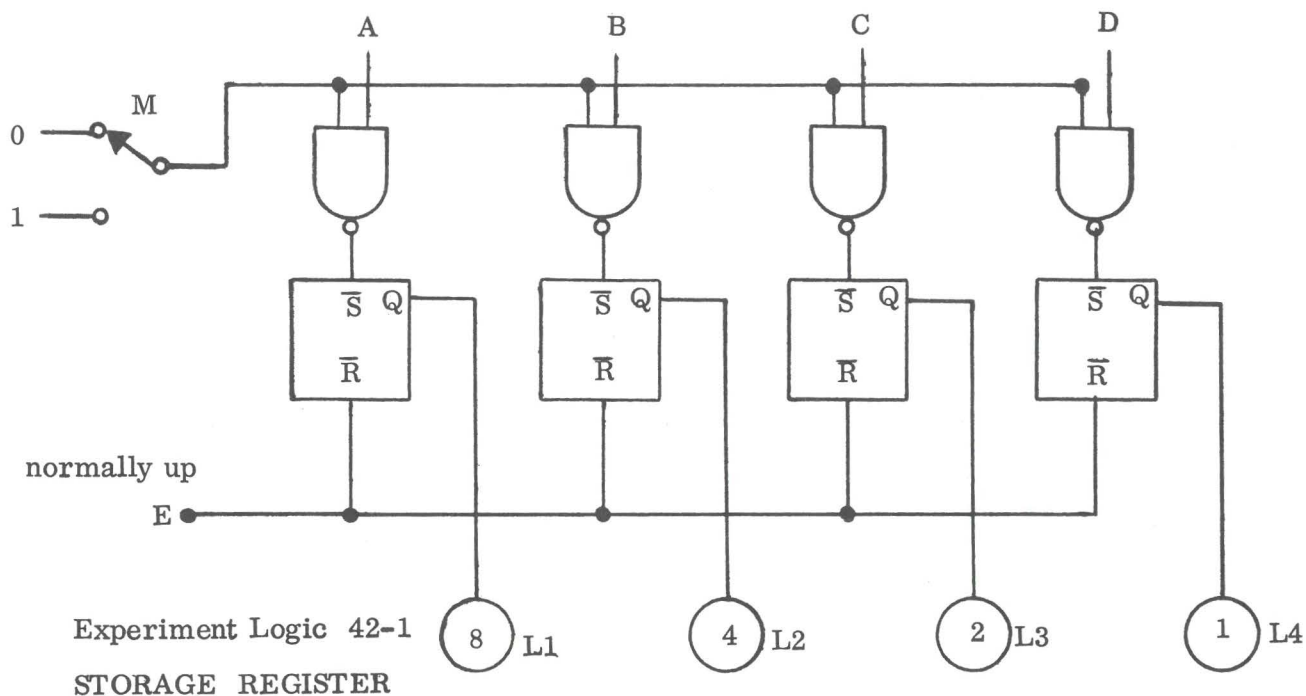
6. Draw the logic for $L = ABC$ (same as experiment) where A, B, C, can each appear at different times.

OBJECTIVE:

This experiment illustrates how a number of flip-flops may be used to store a binary number and how the binary number can be set into the flip-flop storage register by means of logic elements.

DESCRIPTION:

Four flip-flops store a number as set-in by switches A, B, C, D.



The data is set in only when switch M is depressed, by means of the NAND gates. The register is reset by setting switch E down then up.

PROCEDURE:

Wire the Logic above. Set numbers into the register. Switches ABCD are set for the binary number. Then switch E is set UP then DOWN to clear the register. Then depress M to set the number. The switches (ABCD) can be changed and the original number will still be stored in the register.

[illegible]

QUESTIONS:

the register?

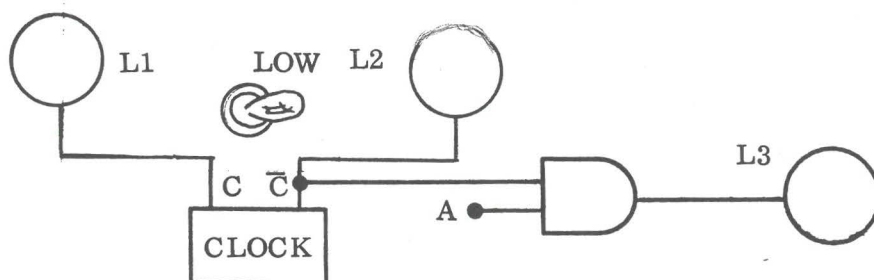
1. What is the maximum binary number that can be stored in
2. How long will the register store this information?
3. What is the advantage of the Storage Register?

OBJECTIVE:

This experiment introduces the clock element and "dynamic operations".

DESCRIPTION:

The clock circuit generates pulses or alternating 1 then 0 then 1 then 0 etc. binary signals. The clock is used to step operations automatically in a computer. When the C output is a "1", the $\bar{C}$ output is "0" and vice versa. The clock signals can be gated (AND) as shown or used in any logical operations. The rate of change of this clock is approximately one pulse per two seconds in the Low speed position and approximately 2,000 pulses per second in the HIGH speed position.



Experiment Logic 43-1

CLOCK

PROCEDURE:

Wire the logic shown. Set switch A at 0. Observe the outputs of L1, L2 and L3 and describe the outputs in the data column. Set switch A at "1" and repeat. Set at the LOW speed position.

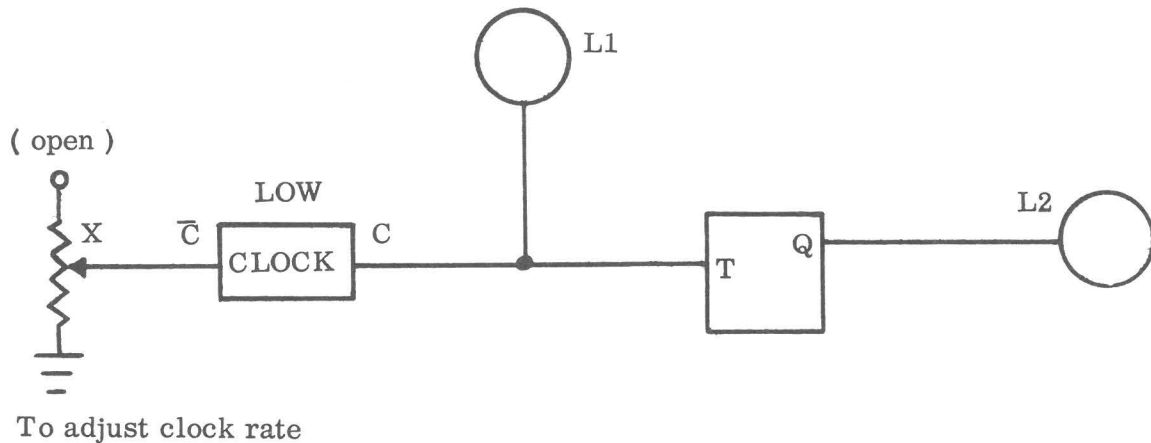
switch A 0	L1
	L2
	L3
switch A 1	L1
	L2
	L3

QUESTIONS:

1. Count the number of "1" pulses in 30 seconds. What is the rate of the clock in "pulses per second" ?
2. What would the output be for an AND gate with inputs C and $\bar{C}$?
3. What is the OR gate output with inputs C and $\bar{C}$?

DESCRIPTION (Triggered flip-flop):

When a clock pulse is connected to the "T" (TRIGGER or TOGGLE) input of a flip-flop, the flip-flop changes from one state to another corresponding to each pulse. The flip-flop output (Q) "changes" when the clock pulse goes from 1 to 0. Therefore, the flip-flop pulses will be half as fast as the clock pulses.



Experiment Logic 43-2
TRIGGERED FLIP-FLOP

PROCEDURE:

Wire the experiment. Observe that the flip-flop output changes when the clock pulse (lamp) goes OFF (to "0"). Write observations in the DATA table. Count the number of C pulses in 30 seconds. Count the number of Q pulses in 30 seconds.

QUESTIONS:

4. How many flip-flop pulses result from 100 clock pulses?
5. How many from 101 clock pulses?
6. How many clock pulses are required to generate 100 flip-flop pulses?

OBJECTIVE:

This experiment demonstrates the use of TRIGGERED flip-flops in implementing a BINARY COUNTER. A series of flip-flops, each triggering the following ones, results in a count of binary numbers.

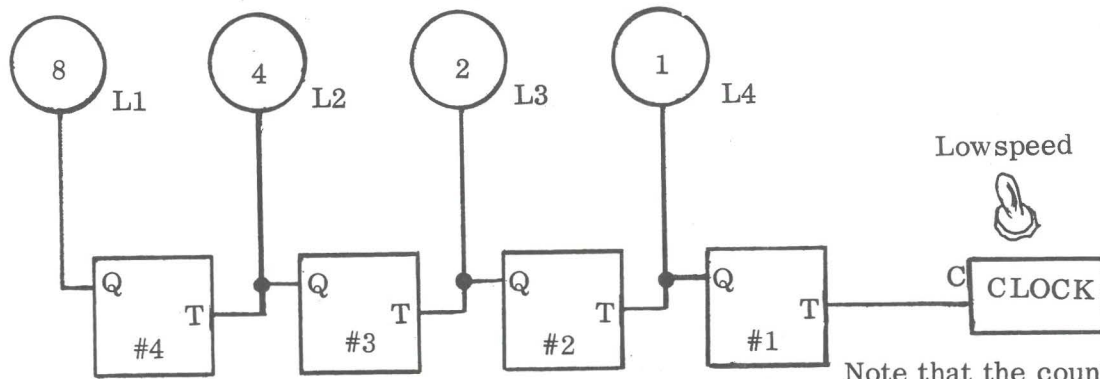
DESCRIPTION:

An observation of the sequence of numbers in the binary number system shows that any a bit always changes when the lower order bit (to the right) goes from a 1 to a 0 (see data table).

BITS

8	4	2	1	N
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9

This corresponds to the operation of the T input to a flip-flop. The series of flip-flops in the logic diagrams , on the following page, is a counter which counts the number of incoming clock pulses (up to 15 and then returns to zero).



Experiment Logic 44-1
UP BINARY COUNTER

Note that the counter schematic is drawn from right to left and the Ed-Lab trainer is wired from left to right.

PROCEDURE:

Wire the experiment. Observe the operation of the counter.

QUESTIONS:

1. How long does it take to count 15 pulses?
2. Draw the logic diagram for a counter which counts up to 63.
3. Draw the logic diagram for a counter which steps the count once for every 1 second (i.e. a binary count of 30 in 30 seconds).

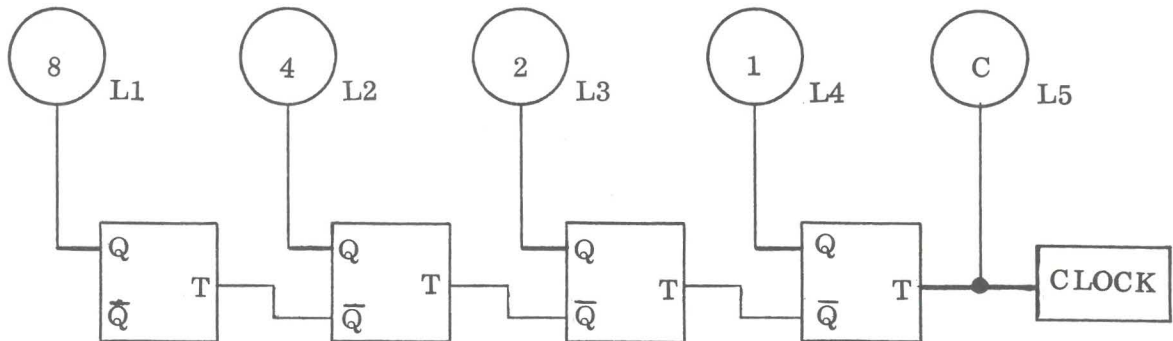
DESCRIPTION (Down Counter):

If the $\bar{Q}$ output from a flip-flop is connected to the next flip-flop, then the result is a down count. That is, if a binary bit (flip-flop stage) changes for a "0" to a "1", the count is down. Observe the table below:

8	4	2	1	N
1	1	1	1	15
1	1	1	0	14
1	1	0	1	13
1	1	0	0	12
1	0	1	1	11
1	0	1	0	10
1	0	0	1	9
1	0	0	0	8
0	1	1	1	7
0	1	1	0	6

PROCEDURE:

Wire the logic. Observe the down-count operation.

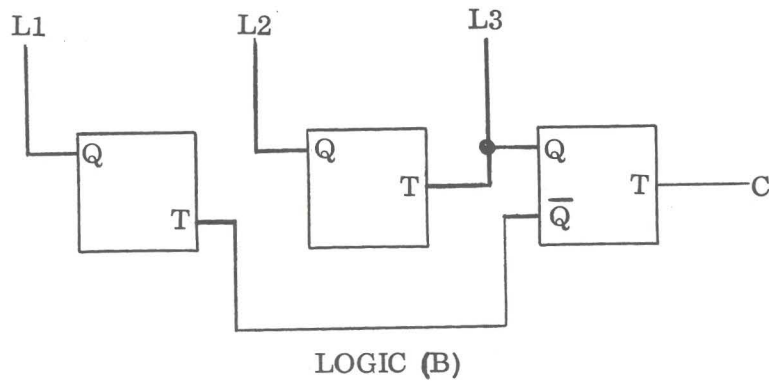
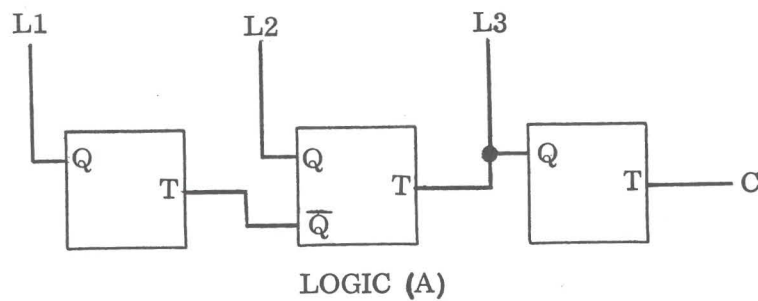


Experiment Logic 44-2

DOWN COUNTER

QUESTIONS:

4. How long does it take to count down by 15 pulses?
5. Write the counting table for Logic (a).
6. Write the counting table for logic (b).



QUESTIONS:

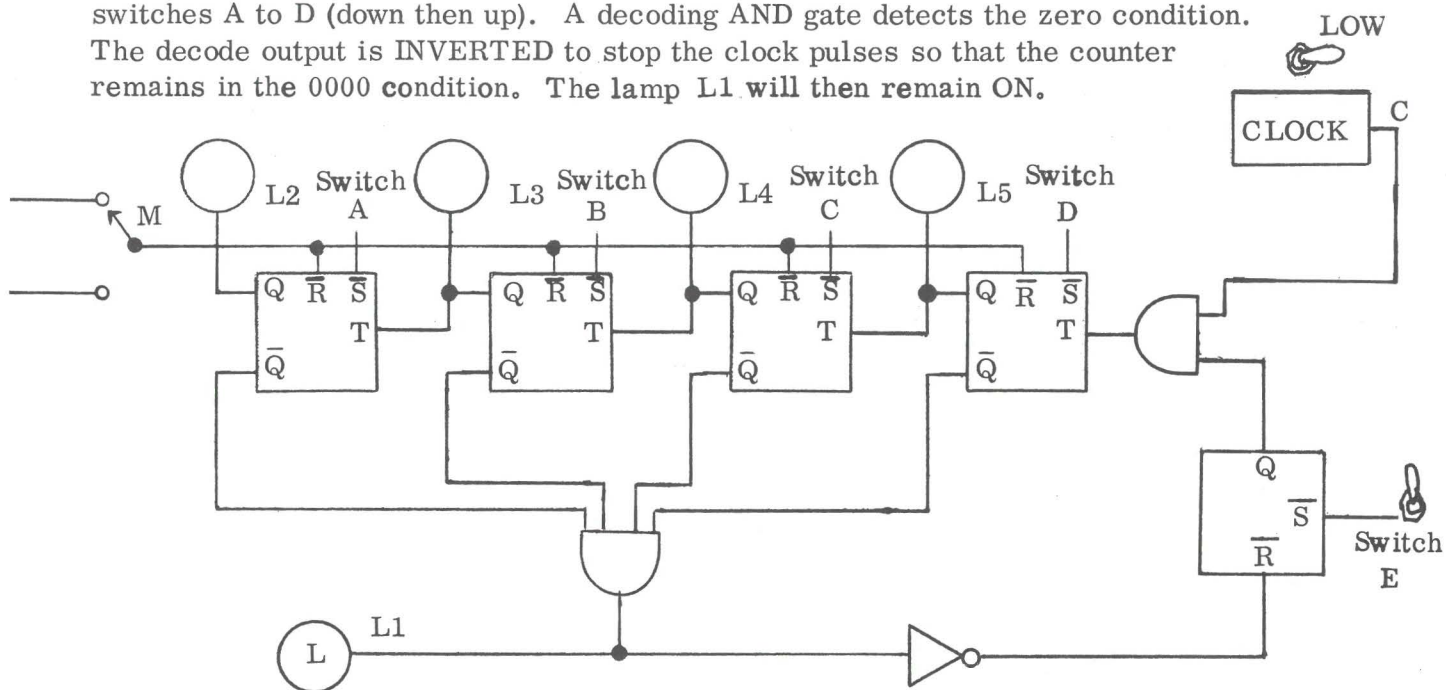
1. After the lamp first " blinks- ON" in ten counts, how many counts will it take until the next " blink " ?
2. Draw the logic gate inputs to decode a count of 12.
3. Draw the logic gate inputs to provide a signal after 8 counts.

OBJECTIVE:

This experiment uses an UP COUNTER to count a present number of pulses.

DESCRIPTION:

The UP counter input trigger is controlled by an AND gate via switch E, as well as decoding logic. The counter can be pre-set to any number by first setting all the flip-flops to "0" by depressing the momentary switch M and then setting 1 into the desired flip-flops by activating any of the switches A to D (down then up). A decoding AND gate detects the zero condition. The decode output is INVERTED to stop the clock pulses so that the counter remains in the 0000 condition. The lamp L1 will then remain ON.



Experiment Logic 46-1

PRESET COUNTER

PROCEDURE:

Wire the experiment. Set switch E UP and wait for counter to stop at 0000. Depress M to set all the flip-flops to 0 (A B C & D should all be UP during this operation). One's may now be placed in the desired flip-flops by moving the switches A to D DOWN then UP as desired. Set switch E DOWN, then UP and the counter will count up until zero is reached and then the counter will stop and the lamp will light. Display lamps are connected to each of the flip-flop outputs so that the number in the counter can be monitored. Record the pre-set binary numbers in the columns of the table on the following page, and the time to reach zero (use a wrist watch).

The maximum count is 16 (counts 0 to 15). As an example, if we preset the number (N) at 9, then it takes 7 counts to reach 0 (or 16). The number of counts or clock pulses or seconds is then $C = (16 - N)$.

A	B	C	D	T SECONDS

QUESTIONS:

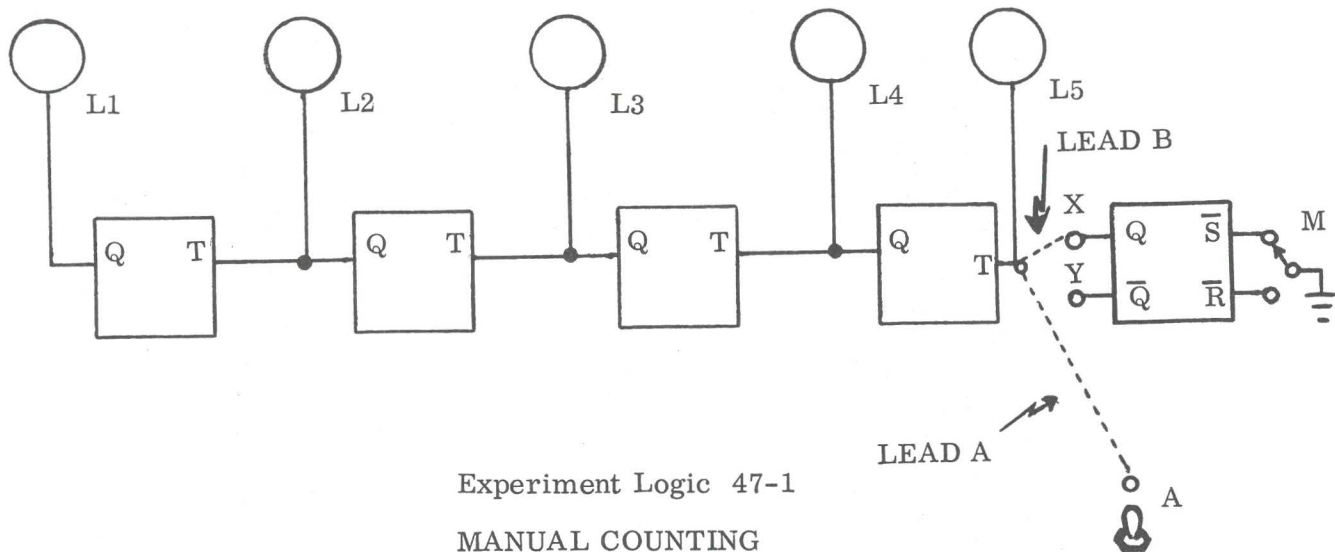
1. What happens if the counter is preset to 0000?
2. List a number of applications for a PRESET COUNTER.

OBJECTIVE:

This experiment demonstrates the requirements for triggering a flip-flop with manual or switch input pulses.

DESCRIPTION:

It was previously indicated that the flip-flop triggered when the input pulse changed from a "1" to a "0". However, this change must be fast and without bounces. A switch is too slow to be used for a trigger and contains many bounces. The switch must be applied to a flip-flop set and clear inputs to generate a fast clean pulse. The counter is triggered from either the Q or $\bar{Q}$ output. This results in a count being added either when the switch is depressed or when it is released.



PROCEDURE:

Wire the experiment. First connect Lead A and observe the counting as A is flipped UP and DOWN. Then connect Lead B to X and switch.M. Repeat for Lead B connected to Y. Record observations:

QUESTIONS:

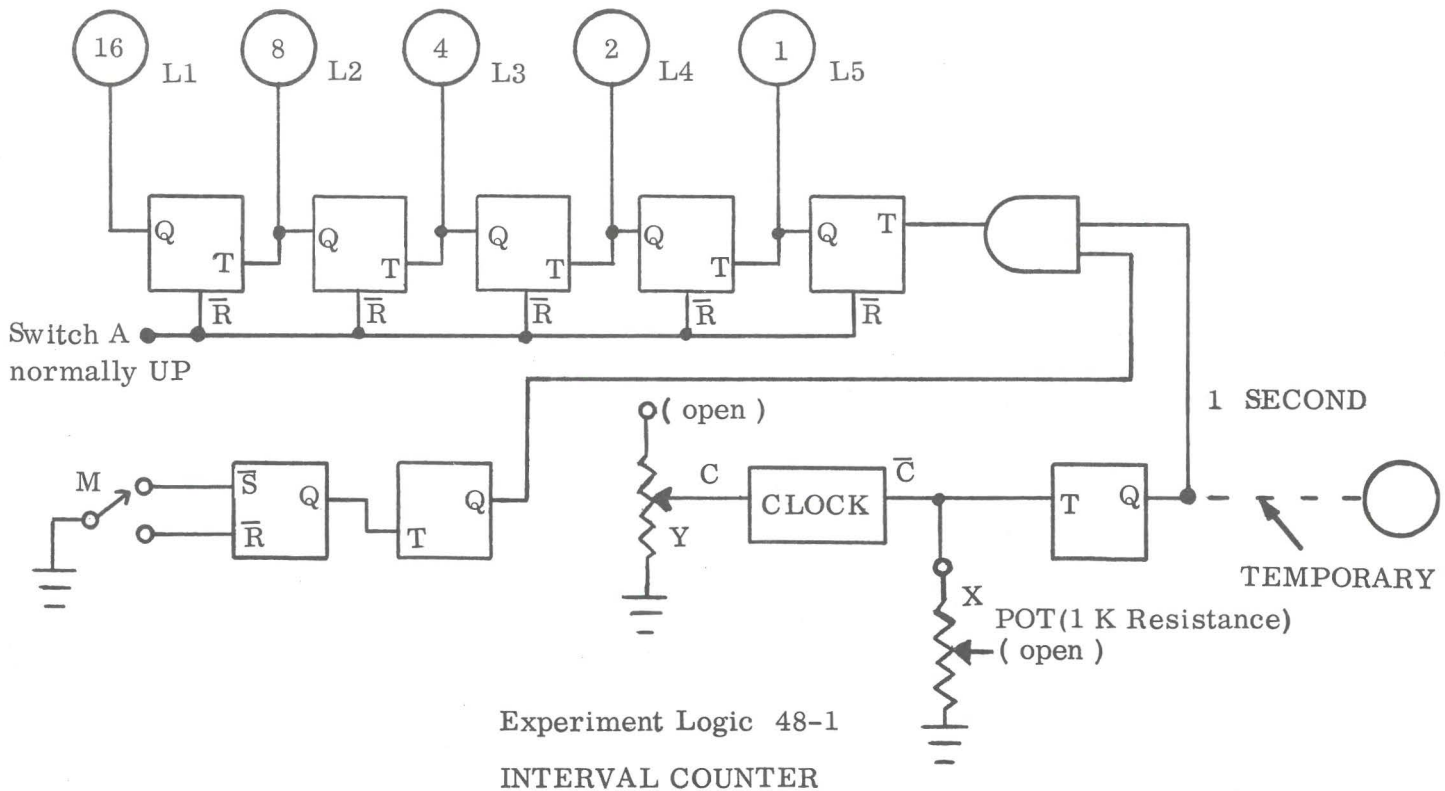
1. When lead A is connected to X, does the counter count on the "depress" or "release" motion or switch M?
2. Use switch A for counting via a flip-flop. Draw the Logic.

OBJECTIVE:

This experiment describes the operation of an INTERVAL COUNTER which measures time duration between two input signals.

DESCRIPTION:

A conventional up-counter is connected to an AND-gated clock input. When the control input (X) is a "0", the counter will not count. When the control input X is a "1", the counter counts the pulses. Switch M sets the control signal to 1 and 0 on alternate impulses of M. The count N is then the time interval (in seconds) between the pulses generated by activating M. This is then a digital STOP WATCH.

PROCEDURE:

Wire the logic. Set A down to clear the counter. Press M if the counter is counting. Then set A down and up to reset the counter to zero. Adjust POT Y for a pulse rate of 1 pulse per second as observed on a display lamp temporarily connected as shown. Depress switch M momentarily then again at a later time. The number N is then the interval between switch M signals. Check the time with a watch and record the data in the table.

TEST	N	Time/watch
1		
2		
3		
4		
5		
6		
7		

QUESTIONS:

1. Calculate the exact pulse rate of the clock using the readings from the watch.
2. How can the "one pulse per second" clock be more accurately set?
3. What is the maximum number of seconds that can be counted using the above system?

OBJECTIVE:

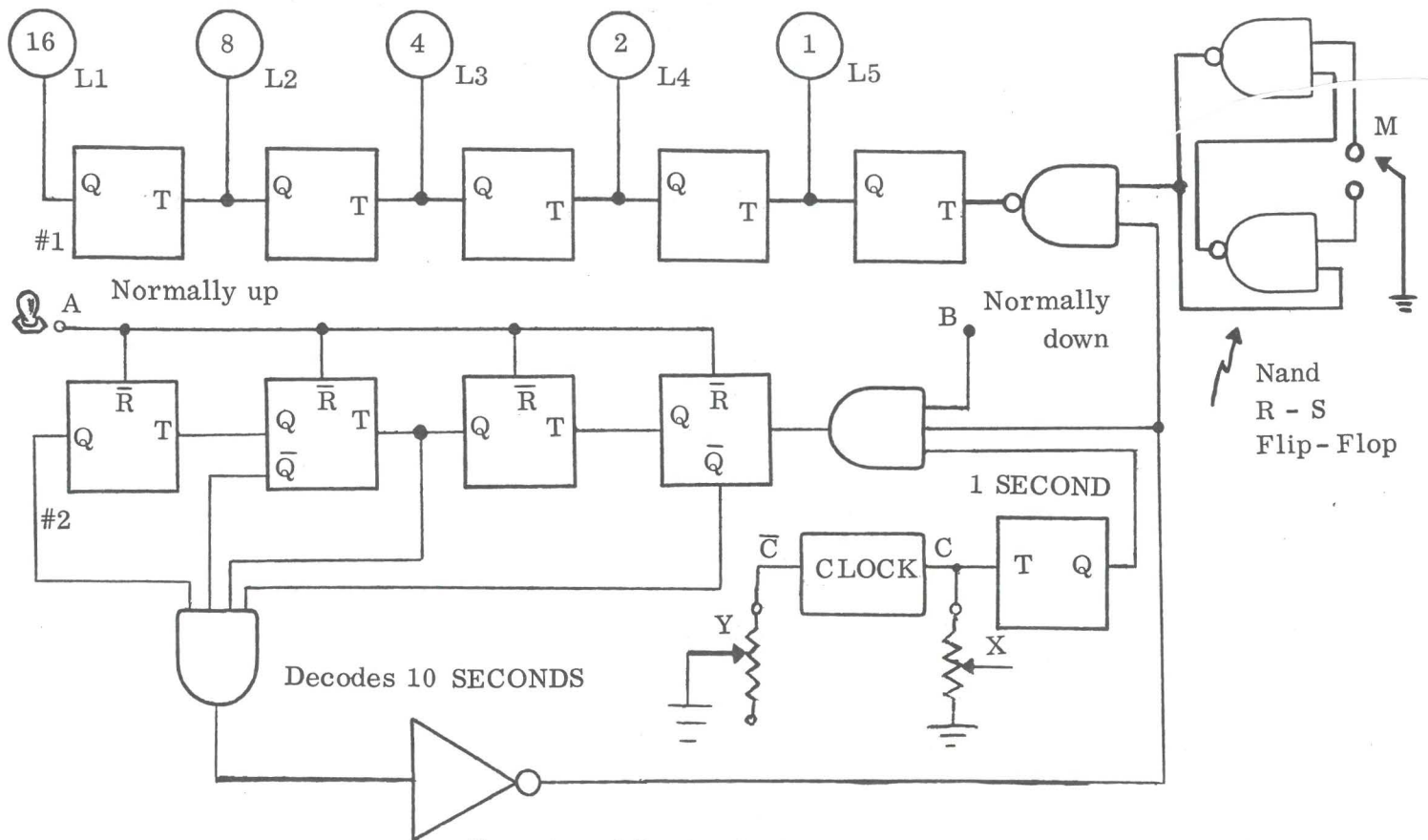
This experiment uses two counters to measure the "rate" of incoming pulses in what is referred to as a "rate-counter" or a "frequency counter".

DESCRIPTION:

Counter #2 receives pulses from the clock when switch B is up. The counter #2 is decoded at a count of 10 for 10 seconds. Counter #1 counts incoming pulses generated manually by switch M. The 10 second decode inhibits an AND gate to halt the incoming pulse count.

PROCEDURE:

Wire the logic shown below:



Experiment Logic 49-1
FREQUENCY COUNTER

Set switch B down. Set A down then up to reset the counter. Use the left hand to depress switch M at various rates (fast to slow). Use the right hand to start count by placing switch B UP. The number read-out is N counts in 10 seconds or N divided by 10 pulses per second. Thus for a count of 12, the rate is 1.2 pulses per second. Enter the data in the table on the following page:

TEST	N	FREQUENCY pulses per second or HERTZ

QUESTIONS:

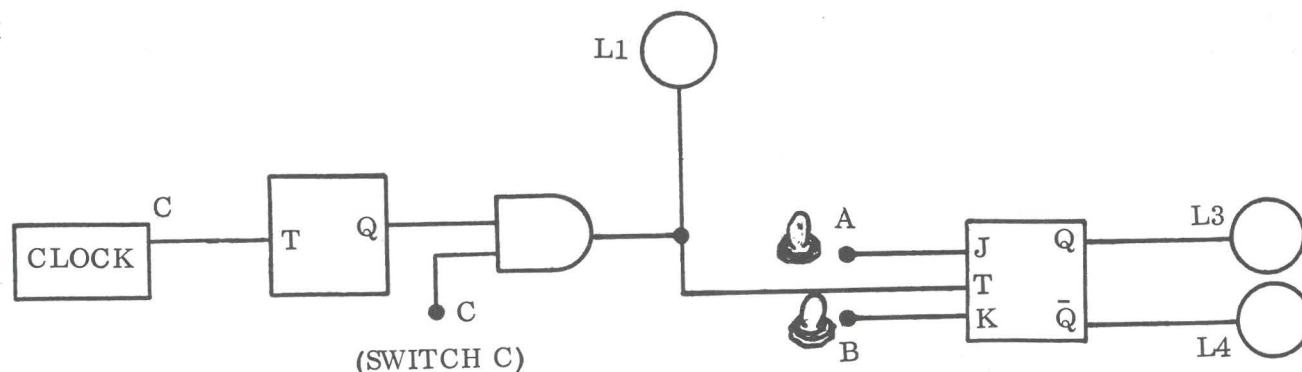
1. How accurate is the count or answer?
2. What basic CLOCK frequency and what count decoding should be used to measure frequencies in the audio range (100 pulses per second to 10,000 pulses per second)?
3. What frequency should the CLOCK be set at to have the counter read directly with a decimal point at 10.101
This fractional number is read as ($2 + 0.5 + 0.125$) 2.625 pulses per second.
Adjust the frequency and run the test.

OBJECTIVE:

This experiment demonstrates the use of the J and K inputs which direct or "steer" the trigger input T to set ($Q=1$) or clear ($Q=0$) the flip-flop.

DESCRIPTION(Steering):

The steered inputs J and K are set-up by switches A and B. Each can be set up as a "1" or "0". There are thus 4 combinations of inputs. With switch C down there is no input trigger clock pulse. With $J=1$ and $K=0$ a trigger clock will set the flip-flop to $Q=1$. With $J=0$ and $K=1$, a trigger clock will CLEAR the flip-flop to $Q=0$. With both $J=1$ and $K=1$ the output will change for each input trigger. With both $J=0$ and $K=0$ the output will not respond to an input trigger. Without an input trigger ($C=0$) the flip-flop will not change states as J and K are changed.



Experiment Logic 50-1

STEERED INPUTS

PROCEDURE:

Set the inputs in accordance with the DATA table and tabulate the results.

A	B	switch C	Q	$\bar{Q}$
0	0	UP		
0	1	UP		
1	0	UP		
1	1	UP		
1	1	DOWN		
0	1	DOWN		
1	0	DOWN		
0	0	DOWN		

QUESTIONS:

1. What is Q when J=1 and K= 0?
2. What is Q when J=1 and K=1?
3. What is Q when J=0 and K=0?
4. Draw the logic for a trigger-steered flip-flop where the output follows the input of a single switch A.

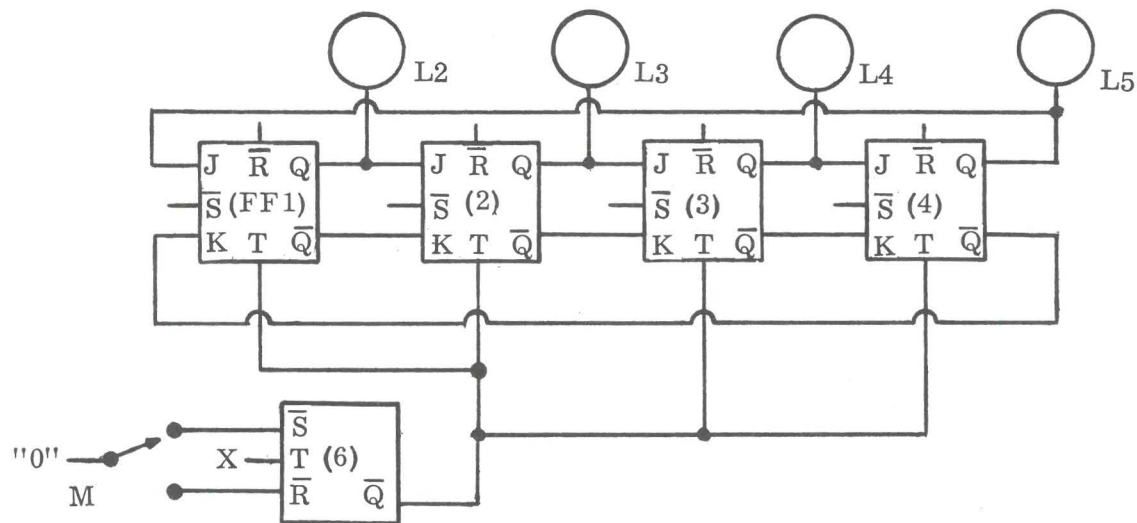
DESCRIPTION (Shift-Register):

The outputs (Q and $\bar{Q}$) of a flip-flop are connected to the steering inputs (J and K) of the following flip-flop. When a trigger pulse (T) is applied to this second flip-flop, the output follows the inputs from the previous flip-flop. When a series of flip-flops are connected, the binary number shifts with each input pulse.

PROCEDURE:

Wire the experiment. Set the flip-flop register to any number by using a lead connected to a "0" (GROUND) and touching the other end to the $\bar{S}$ or $\bar{R}$. By depressing switch M, a "shift-pulse" is transferred to all stages, and the number shifts. Set various numbers and observe the shifted outputs. the CLOCK to(X) input, disconnect M, and observe automatic shifting.

Disconnect the M to "0" switch lead. Connect the CLOCK (at LOW SPEED) to the X (Trigger input of Flip-Flop 6), and observe the automatic shifting of data "around" this SHIFT REGISTER.

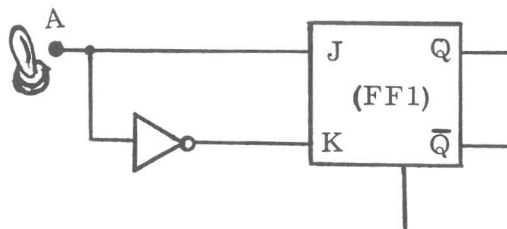


Experiment Logic 50-2

SHIFT REGISTER

EXERCISES:

5. Disconnect lamps L2, L3, and L4. Set in an arbitrary number. Determine the number by using 4 shifts and reading only L5.
6. Instead of feeding the output of the last stage (right hand) to the input of the first stage, connect switch A as input to J and K (use one NOT). Using M, shift IN a 4 bit word, and using only Lamp L5, read out the word.

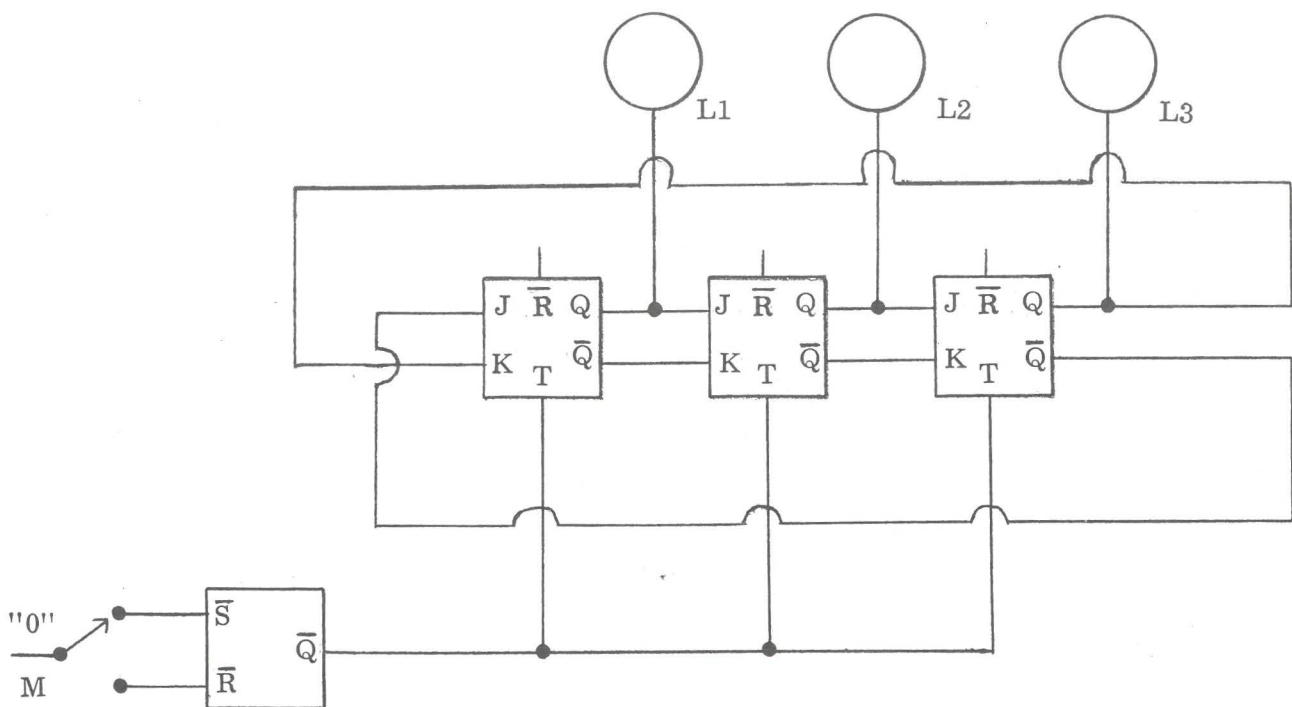


OBJECTIVE:

This experiment illustrates the use of a shift register to generate counting functions. The configurations presented are called: a JOHNSON COUNTER, a RING COUNTER, and a FOUR PHASE CLOCK.

DESCRIPTION (JOHNSON COUNTER):

If the output of the register (flip-flop on the right) is connected to the first flip-flop with the leads reversed ($\bar{Q}$ to J and Q to K), then the bits reverse as they are shifted around to the front of the counter. A three-bit register will have 6 counting positions. A four bit register has 8 counting positions.



Experiment Logic 51-1

JOHNSON COUNTER

PROCEDURE:

Wire the experiment. Set the register to 000 with a lead input from "0" connected to $\bar{R}$. Shift the register and record all outputs.

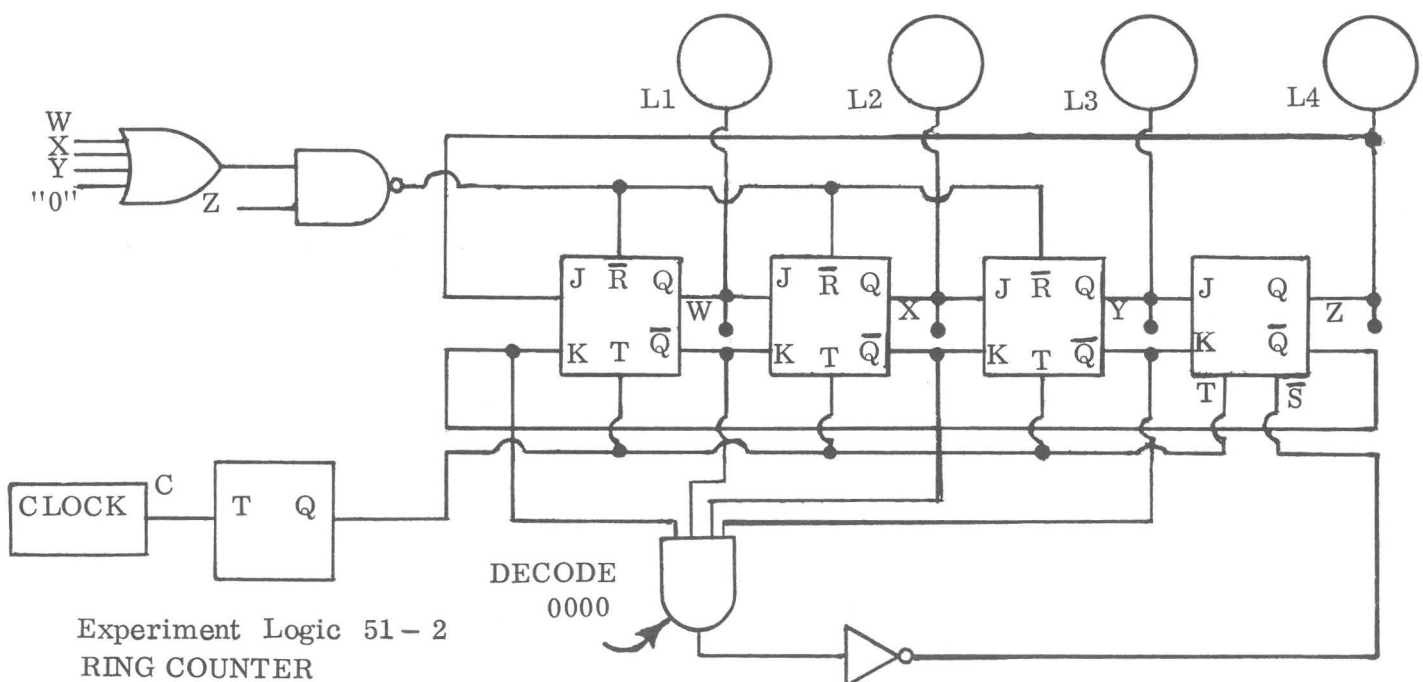
STEP	L1	L2	L3
START	0	0	0
1			
2			
3			
4			
5			
6			
7			

EXERCISES:

1. Determine which binary numbers are not included in the sequence. Set in one of these numbers and record the sequence generated by shifting.
2. Draw the register sequence if the output from flip-flop 2 to flip-flop 3 were also reversed with a count starting from 000.
3. How many states are there for a 4 flip-flop register? Tabulate them. There are two such sequences.

DESCRIPTION (RING COUNTER) :

A conventional shift register is used with an "end-around" connection.



An AND gate determines a 0000 condition and pre-sets a 0001 or a single "1" on the register. When a "1" appears at L4, it automatically clears L1, L2, and L3 flip-flops for the case where more than a single 1 is in the register. These gates maintain the proper register setting.

PROCEDURE:

Wire the logic. Enter the register counts in the data table.

STEP	L1	L2	L3	L4

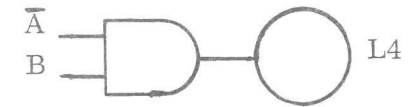
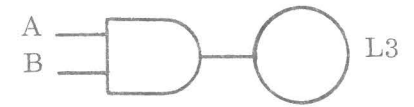
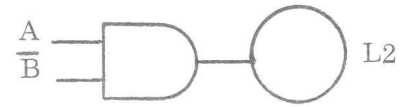
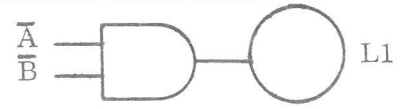
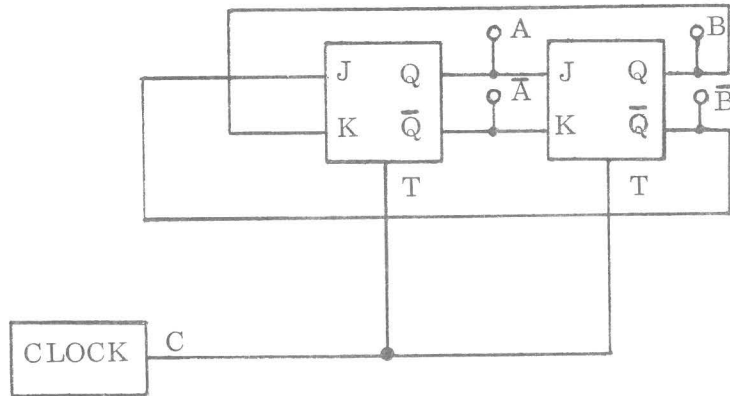
QUESTIONS:

4. What conditions could occur if the AND gate was not included?
5. What conditions might occur if the OR gate was not provided?
6. Draw the logic for a Ring Counter with a single 0, i.e. 1110. Include the resetting gates to assure the proper counts.

DESCRIPTION (FOUR PHASE CLOCK):

A 2 stage Register Counter has 4 count positions. Each position is decoded to generate 4 pulses in sequence. In computer systems, these pulses are used to sequence the steps in the computer operation.

Register Counters
Experiment No. 51 (page 4)



Experiment Logic 51 - 3
FOUR PHASE CLOCK

PROCEDURE:

Wire the logic. Record the observations in the data box.

STEP	L1	L2	L3	L4

EXERCISES:

7. Draw the logic for a 3 phase system with one pulse twice as wide as the other two. The pulses would then be 2 sec, 2 sec, 4 sec, repeat.

8. Wire these gates and check the operation.

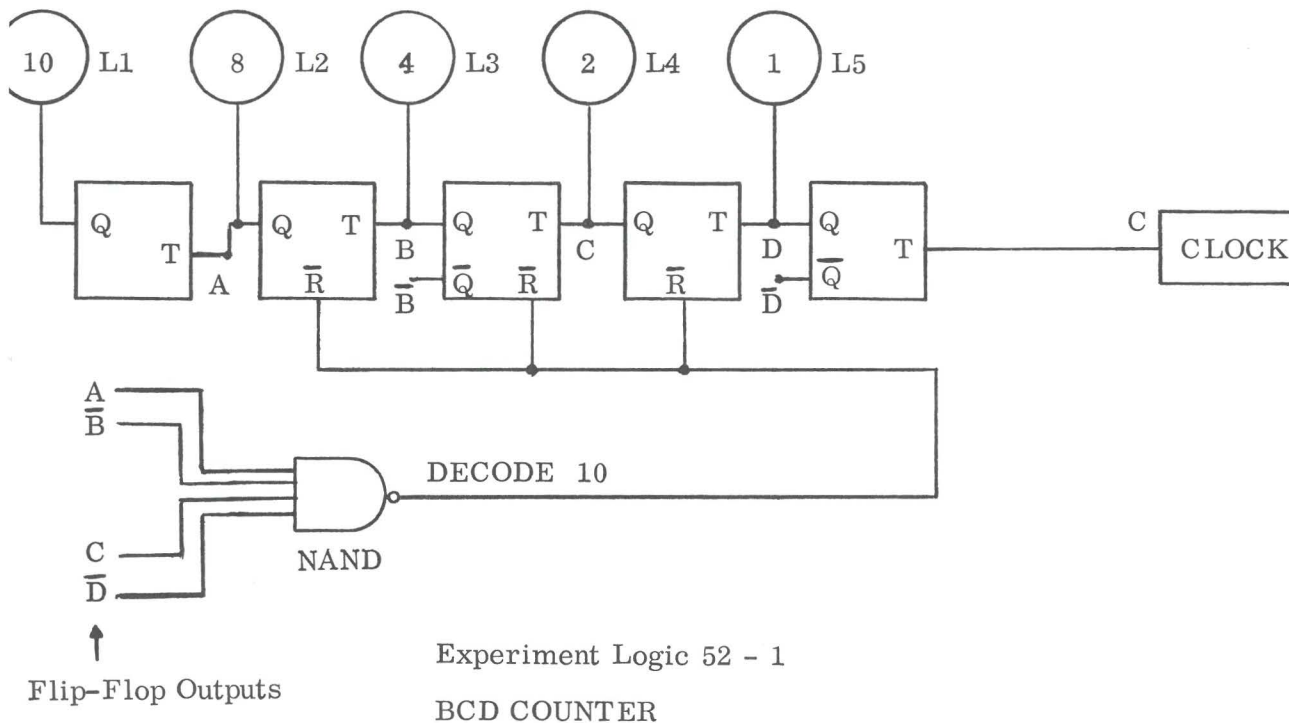
9. The outputs A and B give a TWO PHASE CLOCK. Prepare a table for these two outputs.

OBJECTIVE:

This experiment demonstrates a counter which is reset at a count of ten, so that it counts in a binary coded decimal or BCD manner.

DESCRIPTION:

A conventional UP-COUNTER is triggered from the clock. A count of 10 is DECODED to reset the counter to zero. The counter output also provides a CARRY or OVERFLOW output to the ten's decimal stage.



PROCEDURE :

(1) Wire the logic. Observe the count and record the counts in the data table.

TENS		UNITS			
L1	L2	L3	L4	L5	#

(2) Remove the $\bar{B}$ and $\bar{D}$ inputs from the NAND gate.
 Does the circuit logic still operate?

EXERCISES:

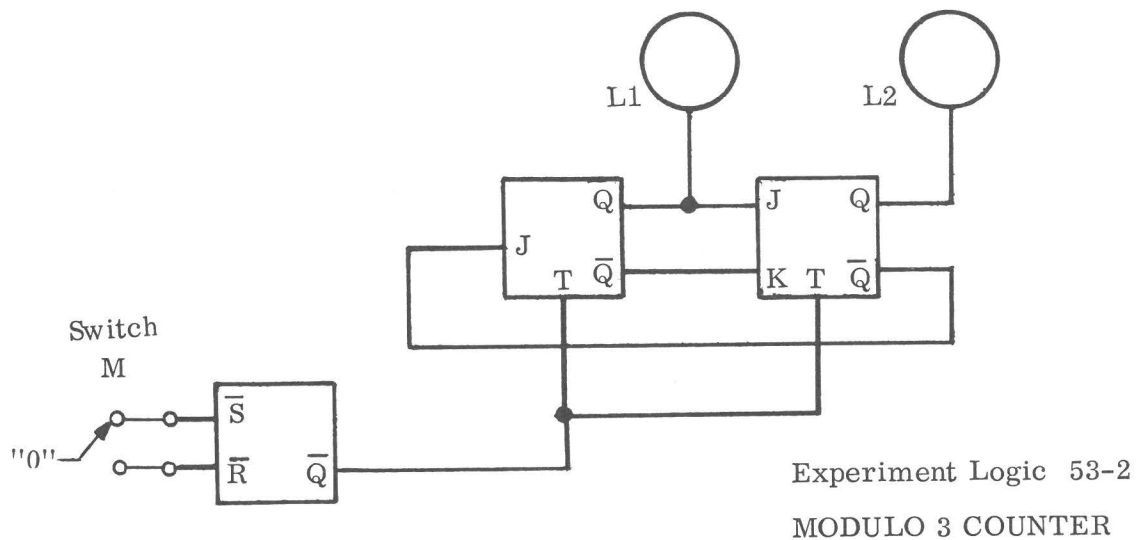
1. Draw the logic for a " duodecimal counter which resets at 12 (i.e. , counts from 0 to 11) .
2. Draw the logic for an octal counter which resets at 8 (i. e. count from 0 to 7) .

QUESTIONS:

1. Which three conditions (there are eight conditions possible from the 3 flip-flops) do not occur in each sequence?
2. What will occur if the register is set into any of these conditions?

DESCRIPTION (MODULO 3):

The shift register steering gates are connected in the appropriate manner to generate a 3 step pattern.



PROCEDURE:

Wire the logic. Record the 3 states in the DATA box. Use switch M to step the register. Review the logic to verify the operation.

[illegible]

QUESTIONS:

3. Draw the logic to provide MODULO 6 counter.
4. Select a random connection of 3 flip-flops using the trigger as well as steering (shift) inputs. Determine sequence.

Switch B UP	
Switch A DOWN	

QUESTION:

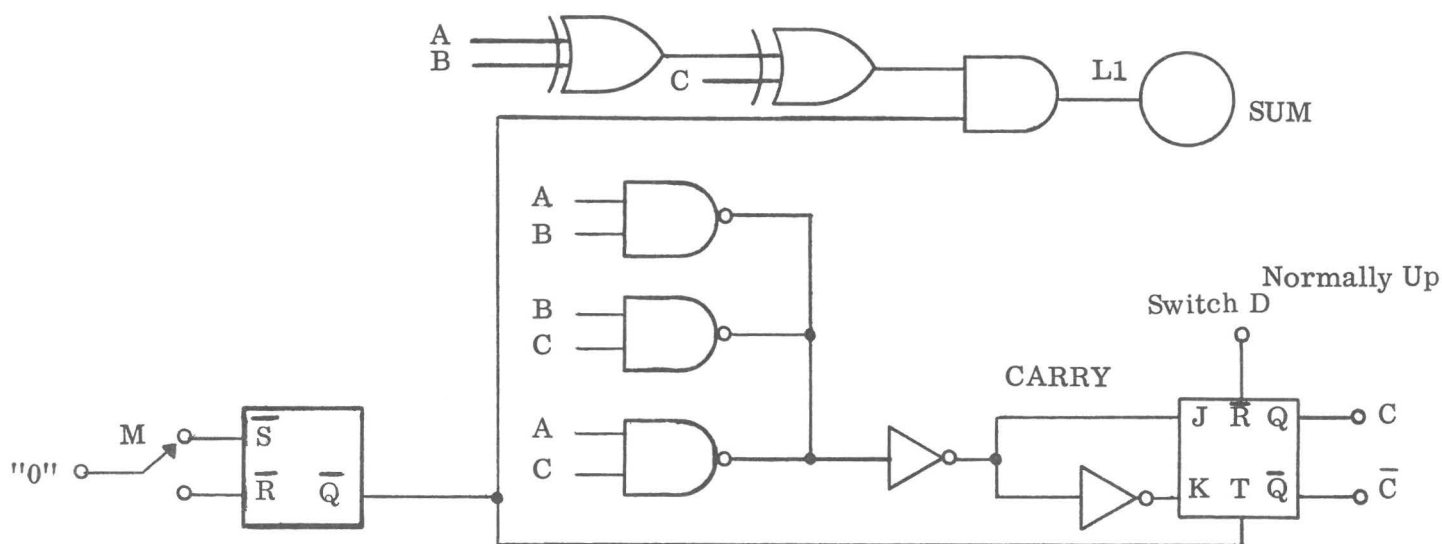
1. What is the cause of the decoding error?

OBJECTIVE:

A FULL ADDER is used with an automatic CARRY memory.

DESCRIPTION:

In a previous experiment using only logic gates, it was necessary for the operator to record the CARRY output from the ADDER and enter the "old-CARRY" data via a switch input. A flip-flop can automatically perform this memory function. The momentary switch M sets a flip-flop and generates a SUM output. CARRY logic is connected to the steering inputs of a CARRY flip-flop. When switch M is released, a TRIGGER pulse (negative pulse) is generated and the new carry information is stored in the memory.



Experiment Logic 55 - 1
ADDER WITH MEMORY

PROCEDURE:

Wire the experiment. Start by clearing the memory (switch D DOWN, then UP). Set switches A and B for the two Binary numbers to be added starting with the least significant, bit (right bit). Depress switch M to read the sum and enter it in the data box. Set up the next A and B digit and continue.

	32	16	8	4	2	1	#
A							22
B							+13
sum							
A							
B							
sum							
A							
B							
sum							

QUESTIONS:

1. What is the logic for a subtractor?

2. Design the logic for a GRAY to BINARY Converter using

MEMORY.

3. Design the logic for a BINARY to GRAY Converter using

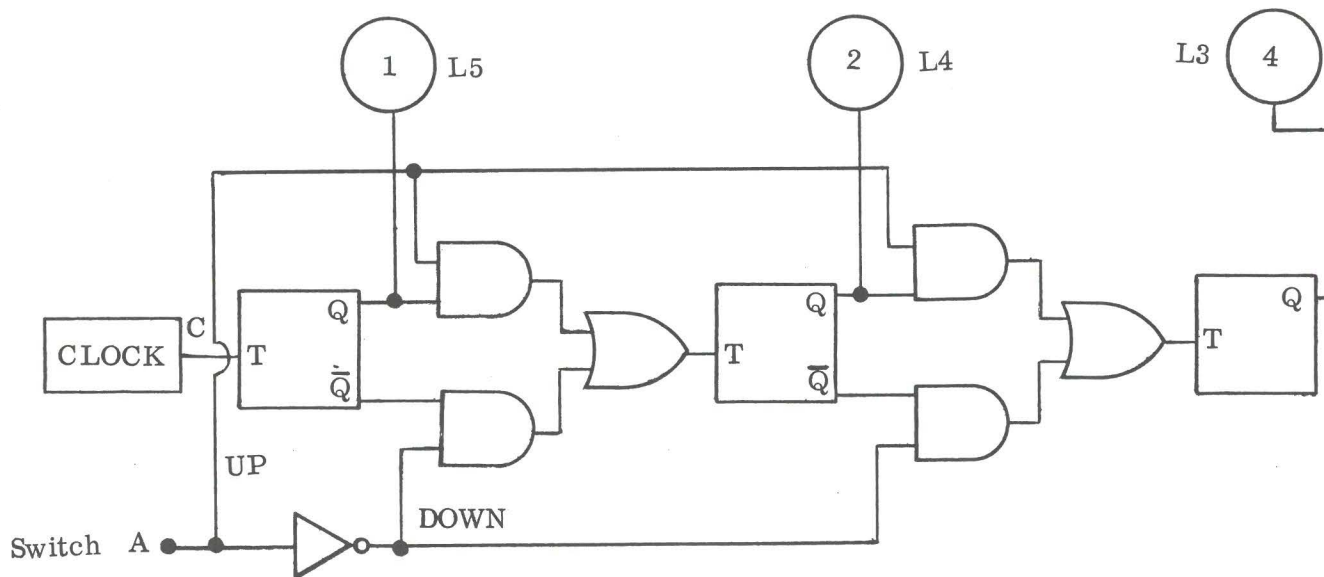
MEMORY.

OBJECTIVE:

This experiment demonstrates the logic for a binary counter which is controlled as an UP or a DOWN counter, and for a SYNCHRONOUS counter.

DESCRIPTION(UP-DOWN Counter):

The conventional binary counter uses the Q output for an UP count and the $\bar{Q}$ output for the down count. Switch A is used to enable AND gates which select the proper flip-flop outputs for the up and down counts. An OR gate is used to OR the two AND gate outputs for the TRIGGER input to the next flip-flop.



UP = add
Down = subtract

Experiment Logic 56 - 1
UP - DOWN COUNTER

PROCEDURE:

Wire the logic. Observe the operation with A-UP and with A-DOWN. Record the observed results.

A UP	
A DOWN	

QUESTIONS:

1. Draw the logic for a shift-register which can shift left or right as controlled by switch A.
2. Wire the right-left shift register. Use the manual switch M to control the shifting pulses. Set numbers into the shift register and observe the operation.
3. Draw the logic for another bit in Figure 56-1 using NAND or other gates available on the trainer.

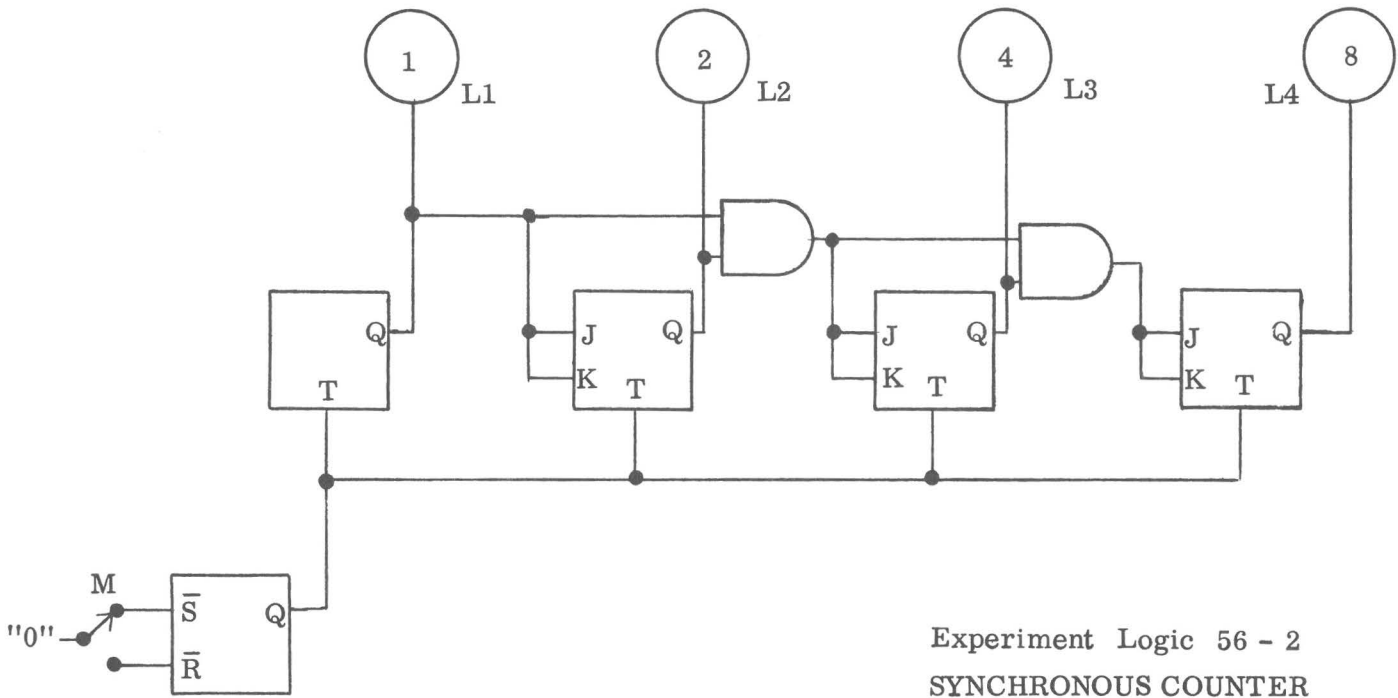
DESCRIPTION (Synchronous Counter):

Previous experiments presented counters where one binary stage triggered the next stage in what is referred to as a RIPPLE COUNTER. Other counters used shift register techniques where all stages change "synchronously". By using logic gating, the count sequence can be controlled with all stages changing simultaneously. This experiment uses gating to produce a binary count.

In an UP binary count, the binary condition of a stage or bit changes on an incoming pulse when the conditions of all previous stages are "1". An AND gate decodes a "1" from all previous stages and provides a 1 to the J and K inputs resulting in a toggle action.

PROCEDURE:

Wire the logic. Step the input and record the output count.



Experiment Logic 56 - 2
SYNCHRONOUS COUNTER

M	L1	L2	L3	L4
1				
2				
3				
4				
5				
6				
7				

QUESTIONS:

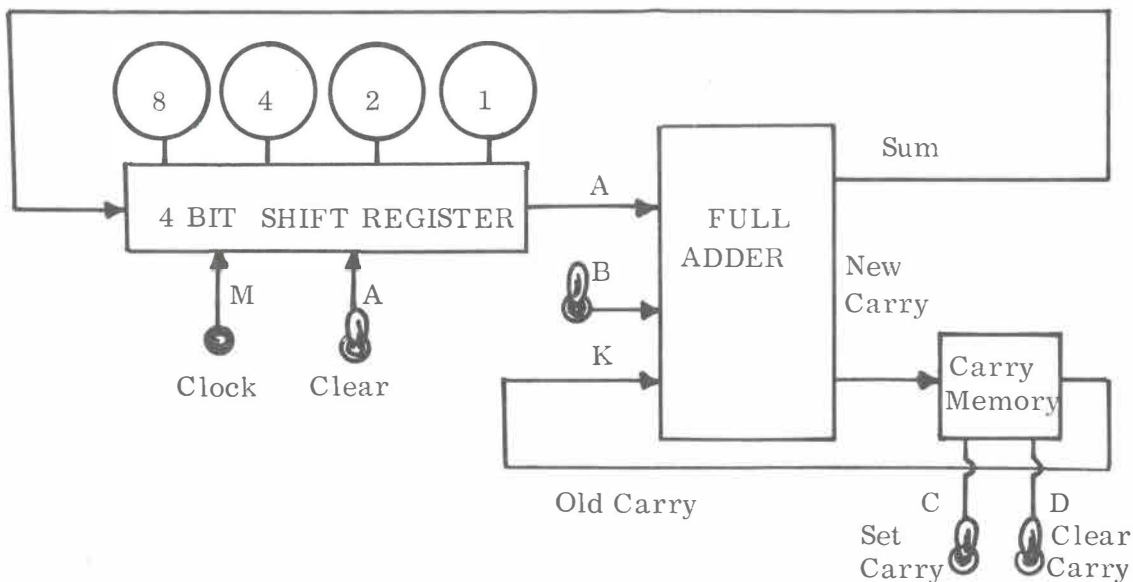
3. Draw the logic for a Synchronous Down Counter.

OBJECTIVE:

This experiment demonstrates the digital system combination of a SHIFT REGISTER and a FULL ADDER and a Data Input Switch to result in an ARITHMETIC UNIT capable of Adding, Subtracting, Shifting, Comparing, and performing other operations similar to those used in Digital Computers.

DESCRIPTION:

The Shift Register and Serial Full Adder were described in previous experiments. A Block Diagram of this experiment is shown below:



Block Diagram

The 4 bit shift register is called the A Register and stores or accumulates the results of the Arithmetic Operations. This register is also called an ACCUMULATOR. The SUM output of the Full Adder is thus connected to the input of the Accumulator. Inputs to the system are provided by the operator using switch B (Up = 1, Down = 0). Data is entered by a CLOCK Shift also generated by the operator using switch M (Down = shift, Up = release momentary switch). Switch A is used to Clear the Accumulator to zero (0000) (DOWN to Clear, then up). Switch C is used to set a "1" or Carry into the Full Adder Carry Memory (Down to set a 1, Up is off). Switch D is used to clear the Carry Memory (Down to clear carry, then Up is off.).

The Lamp Display presents the contents of the Accumulator. Data can be entered into the system; and other numbers can be added, subtracted,

multiplied, compared, etc., and various operations can be performed in a sequence to solve different problems.

PROCEDURE:

1. Wire the Logic in accordance with the Logic Diagram in Figure 57 - 1. Note that Flip-Flop 1 is used to generate the Clock signal from the Momentary Switch M. Flip-Flop 2 is the memory for the carry from the full adder.

The normal position for all switches is UP.

2. Data Entry is performed with the least significant bit first.

Enter the binary number for 13 into the Accumulator.

(binary number 1101)

↑ Enter first.

Procedure:

Clear A Switch A Down, then Up.
Clear Carry (K) Switch D Down, then Up.
(Note Switch B is Down and Switch C is UP.)

Enter (1) Switch B Up.
 Depress M and Release.

(The 1 is now entered and appears in the 8 position)

Enter (0) Switch B Down.
 Depress M and Release.
(A Register contains 0100)

second ——— first

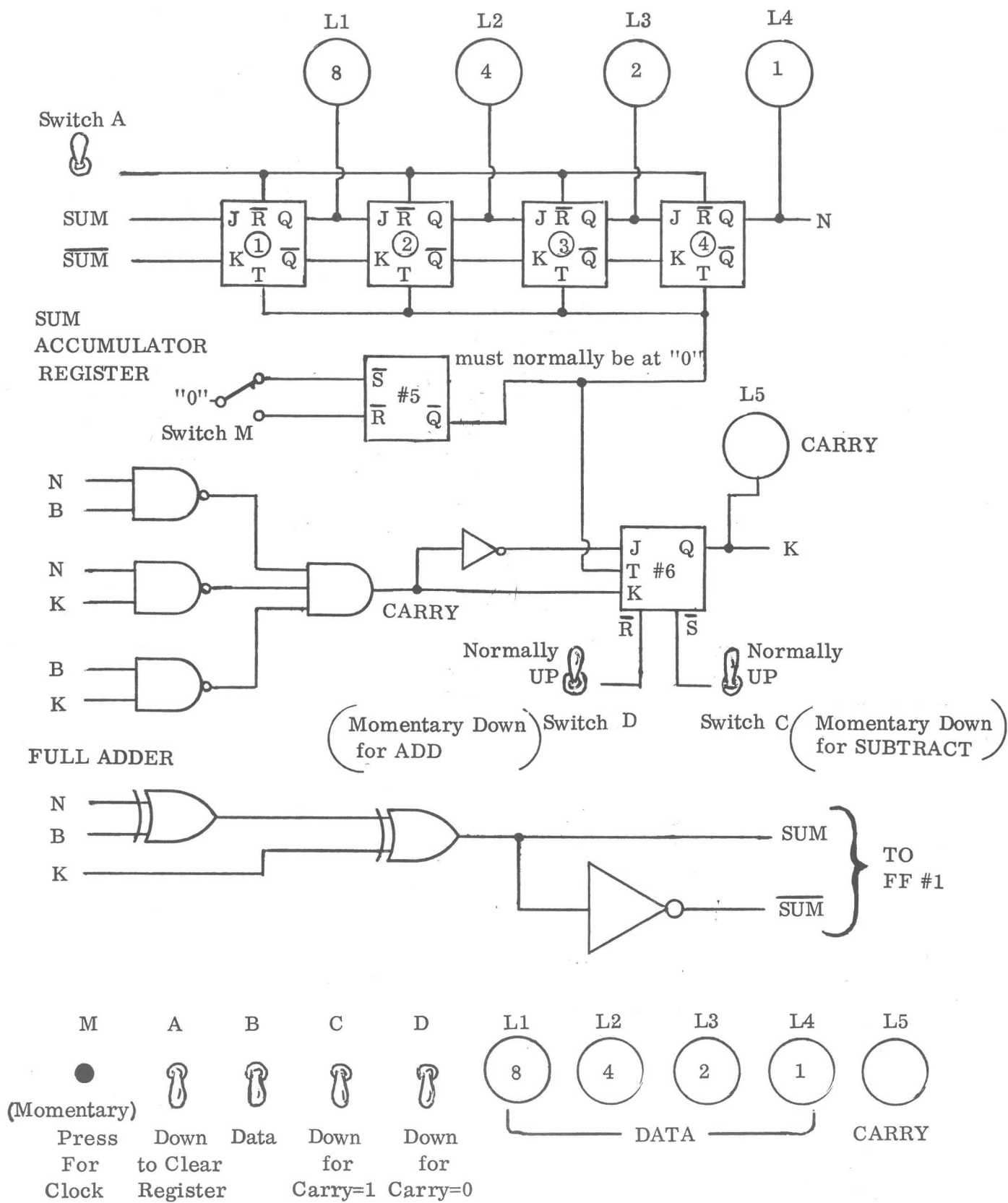
Enter (1) Switch B Up.
 Depress M and Release.
(A is now 1010)

Enter (1) Switch B Up.
 Depress M and Release.
(Display now reads 1101 which is
the data that was entered, No. 13)

Enter the binary number 1101 . Check when complete. _____

3. Enter the following numbers in the register and check when completed.

Binary Number	Decimal No.	Completed
1001	_____	_____
0011	_____	_____
1000	_____	_____
1011	_____	_____
1111	_____	_____
0101	_____	_____



4. Addition is performed by first entering one number. The second number is then entered the same way (WITHOUT THE CLEAR A STEP). Use the Clear Carry C Step. The adder now results in the SUM to be entered into the Accumulator.

Add $6 + 3 =$

Procedure:

Enter 6 (0110)
 Enter 3 (0011) Without the Clear A Step.
 Read the Answer on the display lamps. (1001) = 9 .

Perform the above addition. Enter the result. _____.

5. Add the following numbers.

Decimal	Binary	Answer(Display)	Answer (Decimal)
5 + 5	0101 + 0101	_____	10
5 + 2	0101 + 0010	_____	_____
4 + 9	_____	_____	_____
2 + 12	_____	_____	_____
8 + 9	_____	*	_____

* The maximum permissable answer is 1111 or 15.

Otherwise the answer is an overflow from 16, (16 + 0001).

6. Add the following numbers.

$5 + 2 + 3 =$ _____
 $4 + 3 + 2 + 5 =$ _____
 $4 + \underline{\quad} + \underline{\quad} + \underline{\quad} =$ _____

Enter the binary inputs and outputs and the decimal equivalents above.

Try $3 + 0 =$ _____

7. Negative Numbers are represented by COMPLEMENTS.

As described in previous experiments, the One's Complement of a number is the complement (0 to 1 and 1 to 0) of every bit. Thus, 0011 = 3 The complement 1100 = -3 .

The Arithmetic Unit can generate the One's Complement.

Procedure for One's Complement:

Enter the number (0011) = 3
 Set Carry to 1 Set Switch C Down and leave it there.
 Depress M 4 times Shifts number through adder.
 Clear Carry Switch C UP, Switch D Down then Up.
 Read Answer on display lamps. (1100) = -3

Perform the above conversion. Enter the result. _____.

Note that the most significant bit (the bit on the left) indicates the sign of the number. The maximum number is then 0111 or +7, and the lowest number is 1000 or -7. If we tried to enter number 13, i.e. 1101 in binary, the one's complement number would be interpreted as (0010) so that 1101 = -2 .

8. Complement the following numbers.

Binary	Decimal	Complement	(Decimal)
0011	+3	1100	-3
0101	_____	_____	_____
0110	_____	_____	_____
1011	-4	_____	+4
_____	_____	_____	_____

9. Subtraction in the one's complement system is performed by adding the complement with an end around carry as described in a previous experiment.

Example: 6 - 3 =

	0110	6
	<u>1100</u>	- 3
Sum	10010	
	<u>1</u>	
	0011	+3 Answer

Procedure:

Enter 3 0011
 Ones Compl. 1100
 Add 6, i.e. 0110
 M Clock four times to add end-around-carry.
 Read result 0011 +3.

Perform the above subtraction. Enter the result. _____.

10. Subtract the following numbers:

Decimal	Binary	Answer (Display)	Answer (Decimal)
6 - 2	0110 - 0010	_____	_____
7 - 9	_____	_____	_____
6-4+3	_____	_____	_____
4-2-6	_____	_____	_____
_____	_____	_____	_____

11. The TWO's COMPLEMENT of a number is equal to the One's Complement plus 1 . With the Two's Complement, it is not necessary to add the end-around-carry, and the carry or overflow is ignored and must be cleared prior to the next operation.

Example:

$$\begin{array}{rcl}
 0011 & = & 3 \\
 1100 & = & \text{one's complement} \\
 \underline{1} & & \text{add one} \\
 1101 & = & -3 \text{ in two's complement}
 \end{array}$$

The most positive number is: 0111 = +7 .

The most negative number is: 1001 = -7 .

Negative numbers are converted to positive numbers in the same manner.

Example:

$$\begin{array}{rcl}
 1101 & = & -3 \\
 0010 & = & \text{one 's complement} \\
 \underline{0001} & & \text{add one} \\
 0011 & = & + 3
 \end{array}$$

12. Two's Complement the following numbers:

Binary	Decimal	Two's Complement	Decimal
0011	+3	_____	_____
0101	+5	_____	_____
0100	+4	_____	_____
1011	_____	_____	_____
_____	_____	_____	_____

13. Subtraction in the two's complement system is performed by adding the 2's complement to the number and then clearing the carry.

Example: $6 - 3 =$

0011	+ 3
1101	- 3 (2's complement)
0110	+ 6
10011	sum

ignore carry

0011 + 3 sum

Procedure for two's complement subtraction.

Enter the number (0011) +3
 One's Complement (i.e., set C to 1, shift 4 times, clear C)
 Add one (set C up then down; or use B)
 - 3 is now in register
 Add (+ 6)
 Clear Carry
 Read answer on the display lamps (0011) = + 3

Perform the above subtraction. Enter the result. $6 - 3 =$ (_____)

14. Subtract the following Numbers. Use 2's complement.

Decimal	Binary	Answer (Display)	Answer (Decimal)
7 - 2	_____	_____	_____
2 - 7	_____	_____	_____
3+2-5	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____

EXERCISES:

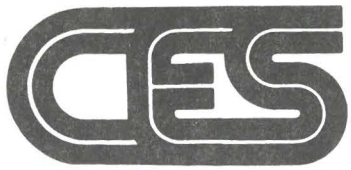
- Draw the Logic Diagram for an Arithmetic Unit with a Full Adder-Subtractor as presented in previous experiments. Use switch C to provide the Carry or the Borrow to the memory flip-flop. Do not use a set C=1 switch. Wire the Logic and perform a number of additions and subtractions.

Try: $5 + 2 - 9 + 3 - 6 =$ _____ .

How do negative numbers appear in the results using the binary subtractor? _____

- Make a table of all numbers in the 2's complement number system from + 7 to - 7.

- Use the Arithmetic Unit to perform Multiplication using the Add and Shift Technique described in an earlier experiment. Use five stages in the accumulator.



Ed-Lab

EXPERIMENT MANUAL

BASIC ANALOG CONCEPTS
CONVERSION TECHNIQUES
COMPUTER CIRCUITS

UNIT NO.

3

CES INDUSTRIES, INC.

EXPERIMENT MANUAL

Copyright © 1974, 1976, 1978, 1980, 1981 CES INDUSTRIES, INC.

Name _____

Course _____

OBJECTIVE:

This experiment demonstrates the means by which electrical voltages are used to transmit or code data for communications or computation. The voltages that correspond to a binary 1 and a binary 0 are first studied. It is then demonstrated how voltages can be varied to correspond to different quantities in what is referred to as Analog data representation.

DESCRIPTION:

The meter on the Ed-Lab panel has three inputs marked +E , +I , and - . When an electrical voltage or electrical power source is connected between the - and either of the other two inputs, and electrical current flows through the meter. This current passes through a coil creating a magnetic field producing a torque or twisting action on the pointer in proportion to that current. The pointer then points to a number on the dial or meter face giving an indication of the current in the coil. This current can be made to correspond to currents, voltages, pounds, feet, etc. When current is applied between terminals marked - and +I , a current of 50 microamperes will result in a full scale deflection. The terminal marked +E has a series 100,000 ohm resistance between itself (+E) and the +I terminal. The meter pointer still deflects to full scale for 50 microamperes. A voltage of +5 volts is required to force 50 microamperes through 100,000 ohms. Therefore, the meter is a voltmeter for inputs applied to the terminal marked +E and - with a full scale deflection of + 5 volts.

The Potentiometers marked X and Y are variable resistance devices with a constant 1,000 ohm resistance and a wiper arm controlled by the knob to position a tap on the resistance. This is used to create or generate different voltages for use in the experiments.

Voltages are used to code the binary numbers 1 and 0 . A 0 voltage is used to code the binary 0 and a positive voltage (approximately 4 volts) is used to code the 1 . Actually, a binary 0 is coded by any voltage between 0 and 1 volt, and a binary 1 is coded by voltages between approximately 2 to 5 volts. The voltages between 1 and 2 volts is a forbidden zone in the binary coding. This experiment will measure the actual binary levels for input and for output on the logic circuits.

Voltages are used in Analog computations to correspond to computation variables. Thus, 5 volts can correspond to 5 feet, 4 volts to 4 feet, etc. Or , 5 volts can correspond to 5 pounds, 3 volts to 3 pounds, etc. Also, 5 volts can correspond to 5 degrees heat. Using a scale factor , 5 volts can correspond to 124 miles, with 4 volts for ___ miles, and 3 volts ___ miles, etc. The scale factor is 5 volts per 124 miles or ($124/5$ miles per volt) 24.8 miles per volt. Thus, 1 volt = 24.8 miles, 2 volts = 49.6 miles, 3 volts = 74.4 miles, 4 volts = 99.2 miles , and 5 volts = 124 miles. Note that 4.4 volts = 109.12 miles. The meter has a maximum range of 5 volts with marks every 0.1 volts. Reading can be made to half a line or to a " resolution " of 0.05 volts. The full scale reading can be divided into ($5 / 0.05$) or 100 parts and thus the resolution is to 1 % of full scale. The reading can thus be resolved to only 1.24 miles so that 4.4 volts = 109 miles.

PROCEDURE:

1. Wire the circuit shown in Fig. 58-1. Note that the (+V) plus voltage is taken from "1" output of the power supply. This supplies a zener regulated voltage of approximately + 3.8 volts. Rotate the potentiometer Knob through the ten marked positions shown in Fig. 58-2 (0, 1, 2, 3, etc. to 10). Enter the knob position and meter voltage in table 58-3 The meter voltages are read as shown in Fig. 38-4. The pointer in position A reads 2.3 volts. The pointer in position B reads 2.55 volts. Then plot the results on a graph as shown in Fig. 58-5. Using a ruler, draw a 'best-straight-line' through the points.

2. Use the voltmeter to measure the voltage at the binary 1 output buss.

Binary "1" buss = _____ Volts

3. Connect the NOT Logic circuit shown in Fig. 58-6. Measure the output Binary 1 and 0 volts.

Binary 1 Output (lamp on) = _____ volts

Binary 0 Output (lamp off) = _____ volts

4. Connect the circuit shown in Fig. 58-7. Vary the Potentiometer and measure the input binary voltages required to activate the lamp and the Inverter (NOT).

Lamp L3 OFF	_____ volts to _____ volts	Binary 0
Lamp L4 ON	_____ volts to _____ volts	Binary 1
Lamp Dim	_____ volts to _____ volts	Forbidden zone

Inverter Output

Output Lamp L4 ON	_____ volts to _____ volts	Input binary 0
Output Lamp L4 OFF	_____ volts to _____ volts	Input binary 1
Output Lamp Dim	_____ volts to _____ volts	Forbidden Zone

5. Connect the circuit shown in Fig. 58-8 to set and reset the Flip-Flop. Vary the potentiometer and use the loose lead alternately on the direct set and reset inputs to determine the range of 0 voltages to operate the Flip-Flop.

Direct Set and Reset Operates from 0 volts to _____ volts.

6. Using a circuit similar to that in Fig. 58-7 for the NOT element, determine the 0 and 1 input ranges for the NOR and for the EXCLUSIVE - OR .

QUESTIONS:

1. If it were desired to code the weight of people in an analog fashion, with an assumed maximum weight of 250 pounds, what scale factor would you select ? What is the resolution using the Ed-Lab Meter ?

2. Comment on the accuracy of readings in the graph plotted as shown in Figure 5.

3. What would you consider to be the valid range of voltage for a binary 1 and for a binary 0 as determined by measurements taken ? Use some slight safety factor.)

4. DECIBELS is the ratio of two voltages presented as a logarithm. What is the dynamic range of the METER in decibels ? Use the ratio of maximum voltage reading to minimum reading increment. See Figure 58-9 .

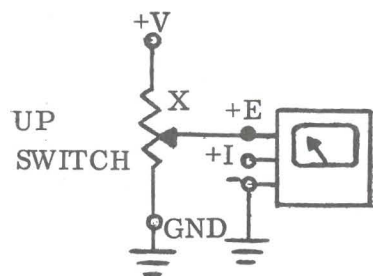


Fig. 58-1 . ANALOG VOLTAGES

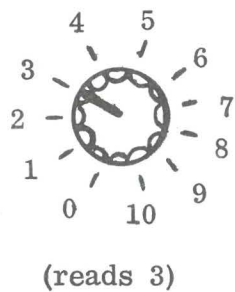


Fig. 58-2 . KNOB

KNOB POSITION	METER VOLTS

Fig. 58-3 TABLE

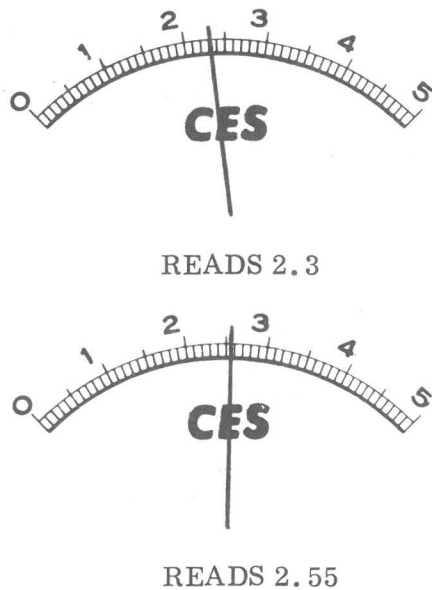


Fig. 58 - 4 METER FACE

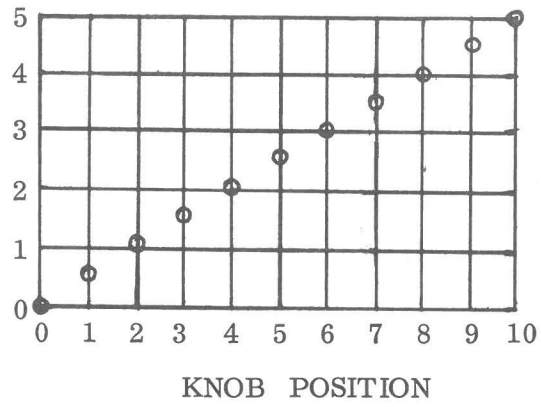


Fig. 58-5a. PLOT

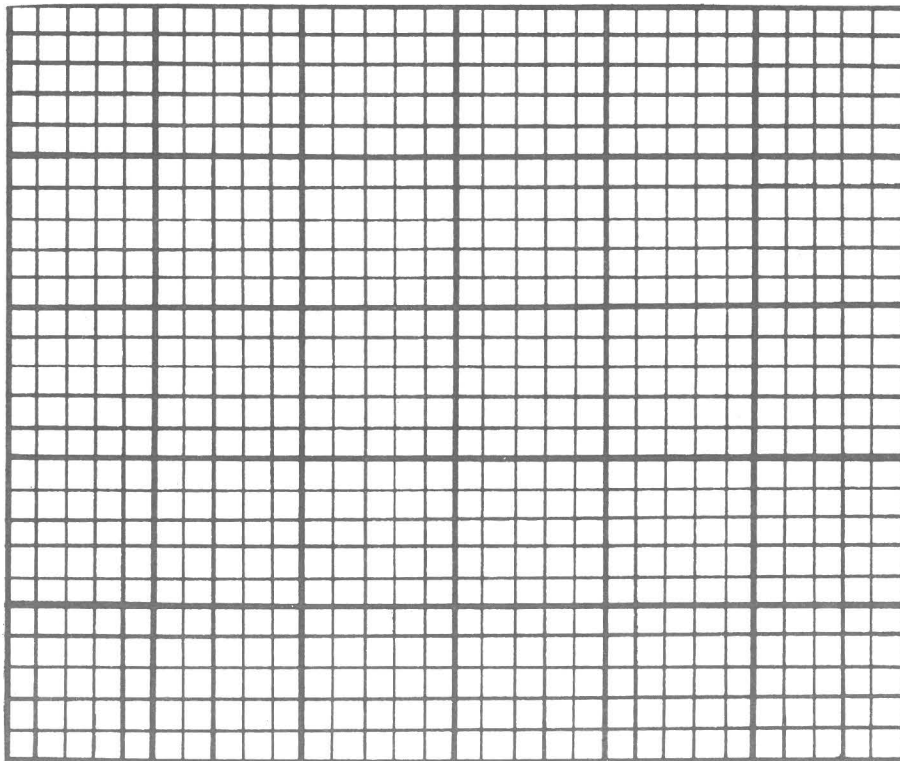


Fig. 58-5b.

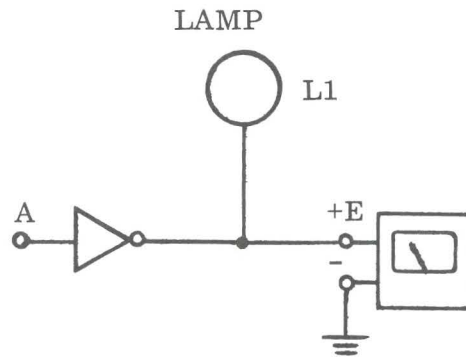


Fig. 58-6. BINARY LEVELS

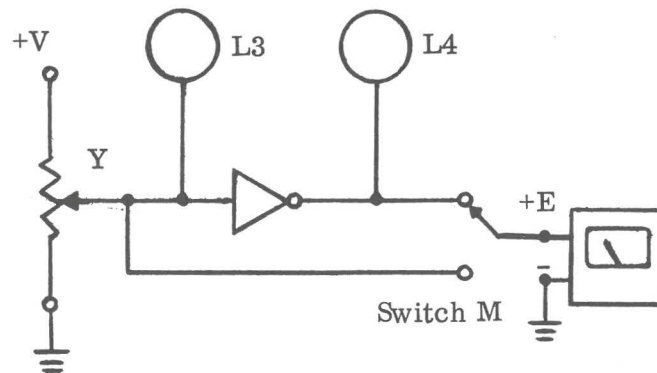


Fig. 58-7. TOLERANCES

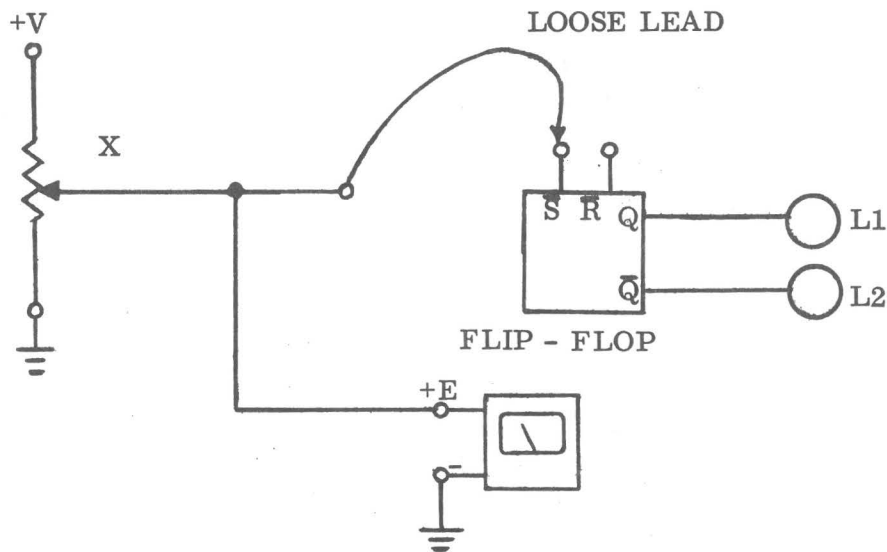
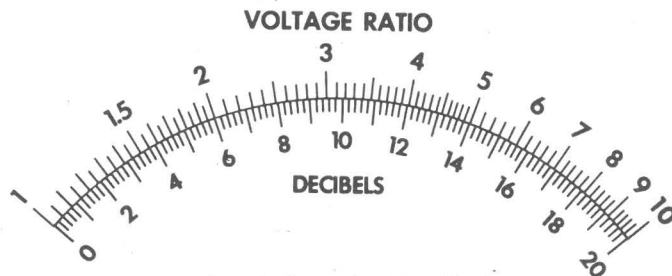


FIG. 58-8 FLIP - FLOP SET/RESET LEVELS



For values of voltage ratios 10^n or 10^{-n} times those shown add or subtract respectively $n \times 20$ DB to or from the values on the DB scale. Examples: for voltage ratios 100 (i.e. 10^2) times those shown, add 2×20 or 40 DB; for voltage ratios 0.001 (i.e. 10^{-3}) of those shown, subtract 3×20 or 60 DB.

For power ratios divide DB scale by 2.

Zero Ref. = 6 mW in 600 ohms = 1.897 volts

DBM Ref. = 1 mW in 600 ohms = 0.775 volts

The scale may also be used as a table of logarithms for rough computations. The mantissa of the log of any number on the voltage scale is equal to 1/20th of the reading on the DB scale. The characteristic of the log of the number is determined in the usual way. Conversely, to find the number corresponding to a given log, multiply the mantissa by 20 and locate this product on the DB scale. The digits of the desired number will be found at the corresponding point on the voltage scale. The position of the decimal point in the desired number is determined in the usual way from the characteristic of the log.

$$\log_e N = 2.3 \log_{10} N$$

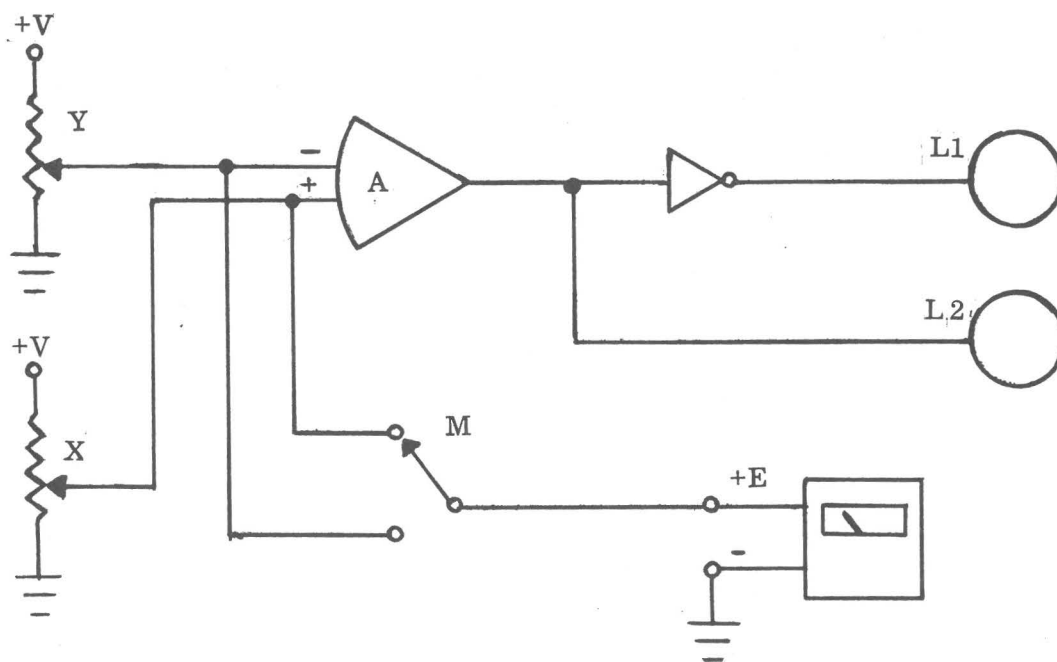
FIG. 58-9 DECIBEL SCALE

OBJECTIVE:

The COMPARATOR (OP-AMP) circuit generates an output "1" when the + input voltage exceeds the - input voltage. This circuit is a COMPARISON device used later in automatic Analog to Digital Conversion.

DESCRIPTION:

The comparator is a differential amplifier, sensing the voltage difference between the + and - inputs and amplifying it so that if + is greater than - the output is a "1", and if - is greater than + the output is "0". In this experiment switch M is used to monitor the voltage levels of the + and - inputs. Lamp L1 will light when the output is "0" and L2 will light when the output is "1".



Experiment Logic 59-1

ANALOG COMPARATOR

PROCEDURE:

Wire the experiment. Set X and Y at various settings and record the values of X and Y and the output "1" or "0".

X	Y		L1	L2

QUESTIONS:

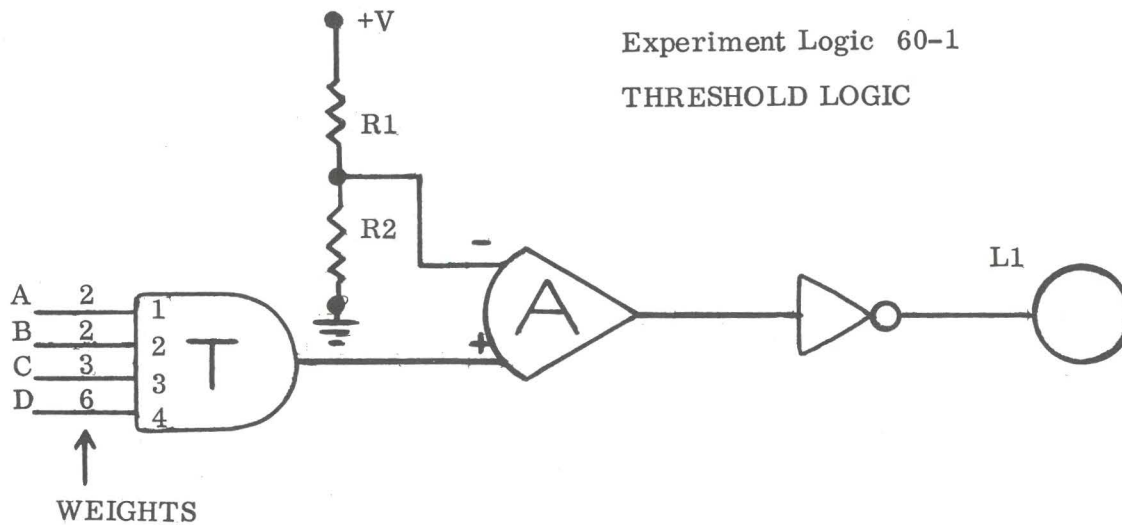
1. How many volts above Y must X be for an output "1" to result?
2. Connect the meter to the output of the comparator and determine the voltage corresponding to a binary 1 and a binary 0.

OBJECTIVE:

This experiment demonstrates a digital logic system which combines analog circuits and the comparator to generate a digital output in what is called THRESHOLD LOGIC.

DESCRIPTION:

The Threshold Logic element is a "resistor-divider" circuit which is connected to the + input of the voltage comparator.



The - input is connected to another divider network. The threshold logic inputs has a different resistance to output. For given selections of inputs there will be various combinations of inputs to result in a "1" binary output. The relative input factors are 2, 2, 3, and 6 as shown above in the diagram. This corresponds to resistors of 15 K, 15K, 10K, 5K ohms. If the combined weight of input 1s exceeds the combined weight of input 0s, the output will be a "1".

PROCEDURE:

Wire the logic. Connect only those switch inputs specified for a given logic function. Use the connections shown in the table and verify the functions.

LOGIC	SET	FUNCTION	CHECK
$\overline{A \cdot B + C (A + B)}$	D open		
$\overline{A \cdot B \cdot C}$	D = 0	NAND	
$\overline{B \cdot C + D}$	A = 1		
$\overline{A \cdot B \cdot C + D}$	----		
$\overline{C + D}$	A = 1 B = 1	NOR	
$\overline{D}$	A = 0 B = 0 C = 0	NOT	

QUESTIONS:

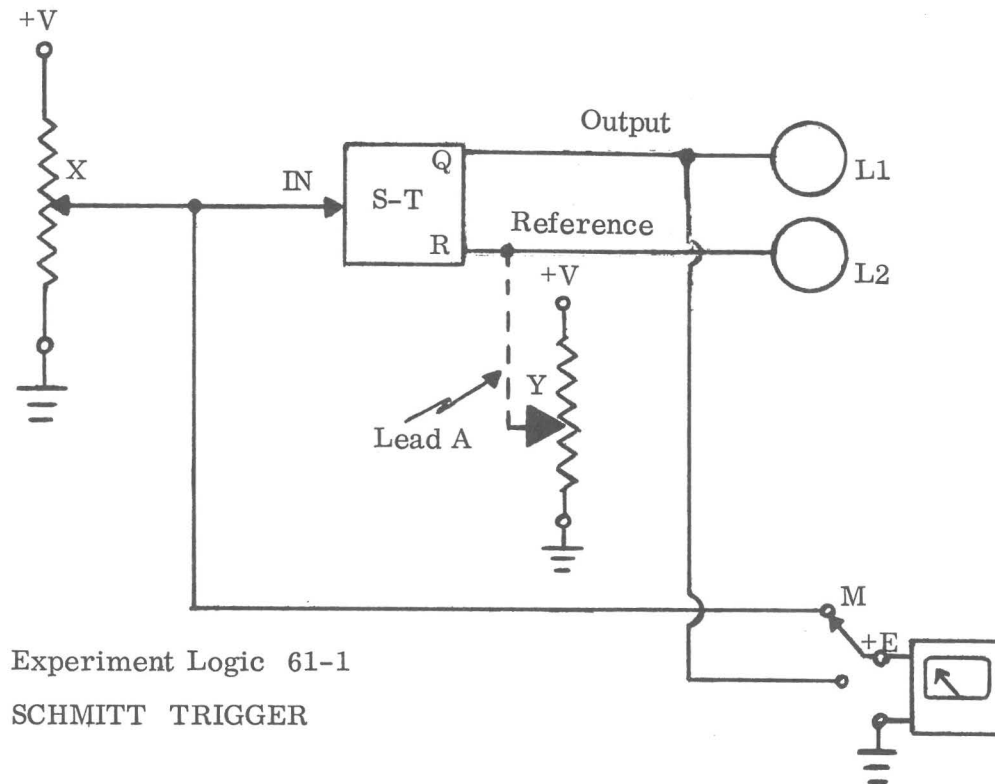
1. Set-up the logic for an AND circuit. (Note: Use T to - and R1/R2 to +).
2. Set-up the threshold logic system for a half-adder. (Assume the threshold logic in a single new logic symbol. Select your own weights.)
3. What are some advantages and disadvantages of threshold logic?
4. Set-up a threshold logic system for a full-adder.

OBJECTIVE:

This experiment demonstrates the operation of a Schmitt Trigger circuit which switches its output state (1 or 0) at preselected input levels. Adjustment of switching point levels and "hysteresis" are also described.

DESCRIPTION:

The Schmitt Trigger is a level detecting circuit. When the input voltage reaches a predetermined voltage, the circuit "flops" into another state. This circuit enables the conversion from analog levels to binary "1" and "0" at predetermined switching points. The use of a potentiometer enables changing the switching level.

PROCEDURE:

Wire the experiment. (1) Disconnect lead A. Vary the input, X, and record the X levels for switching to "1" and to "0". Record the Q output 1 and 0 voltage levels. Note that the X switching levels for an increasing voltage is different from the level for a decreasing voltage. The difference is called the

voltage "hysteresis". (2) The switching level is changed using lead A to the potentiometer. Set Y at different voltage levels and record switching levels and the Hysteresis.

← X VOLTAGE →

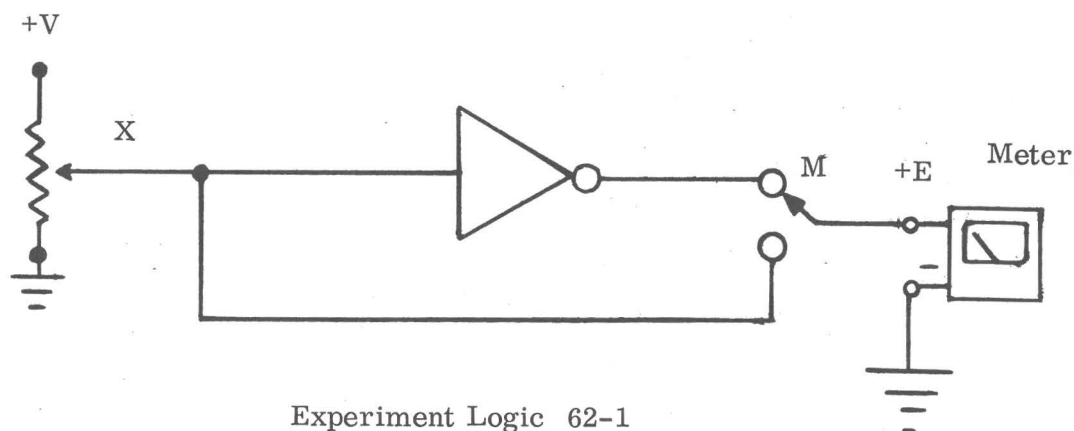
Y VOLTAGE	0 to 1 VOLTS	1 to 0 VOLTS

QUESTIONS:

1. What is the difference between a Schmitt Trigger and a Comparator?
2. Which would be used to determine when an input level exceeded 1 volt?
3. Plot the 0 to 1 switching level as a function of the Y voltage setting.
4. Plot the Hysteresis as a function of the Y voltage setting.

OBJECTIVE:

This experiment uses the digital logic Inverter or NOT circuit as a high gain amplifier with a voltage inversion or negative gain. The gain of the Inverter is so high that there is only a small region of input voltages for which the inverter is not full ON or full OFF.

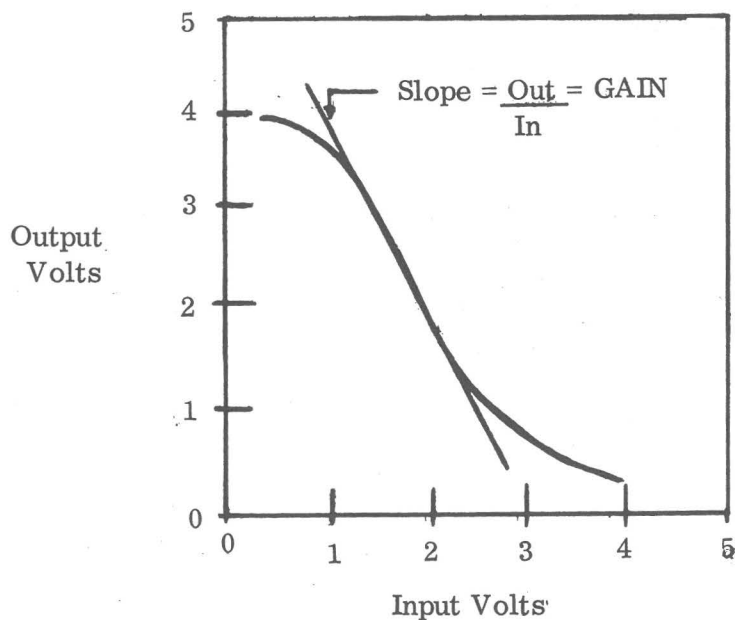


Experiment Logic 62-1

ANALOG AMPLIFICATION

DESCRIPTION:

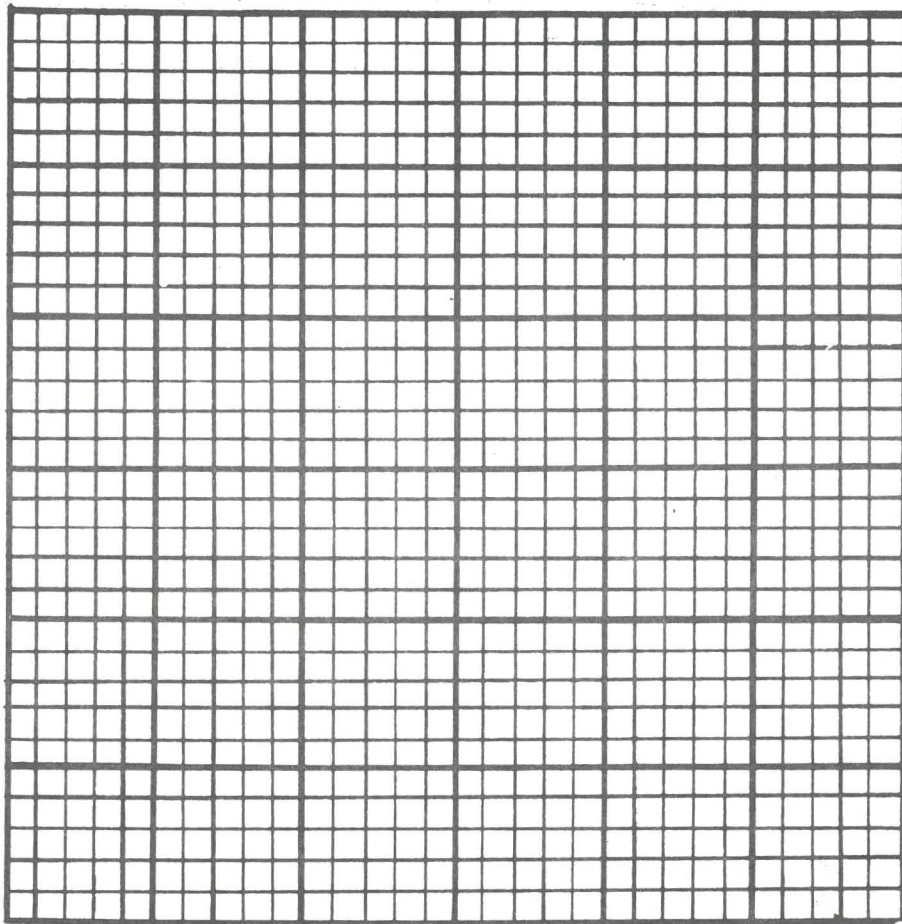
The switch M is used with the meter to read the input and output voltages. The potentiometer X is used to provide a variable input voltage. Thus we can measure and plot the input voltage versus output voltage for the inverter circuit. This plot is similar to that shown below and is called a Transfer Function.



PROCEDURE:

Wire the circuit. Depress M and set the input voltage to zero. Gradually increase the input voltage in 0.1 volt steps and record the input and output voltages in the table below. Plot the results.

[illegible]



QUESTIONS:

1. What is the active region of the inverter circuit (operating input voltage range)?
2. Calculate the gain of the inverter (slope of the curve).

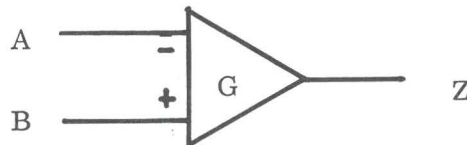
OBJECTIVE:

This experiment demonstrates the characteristics and performance with applications of Operational Amplifiers used in signal processing for computing and instrumentation.

DESCRIPTION :

An Operational Amplifier is a very high gain amplifier used with a portion of the output connected back to the input (known as feed-back) in order to perform various functions of Gain, Inversion, Amplification, Attenuation (Gain less than 1), Addition, Subtraction, Multiplication, Division, Integration, Differentiation, and various other linear and non-linear functions.

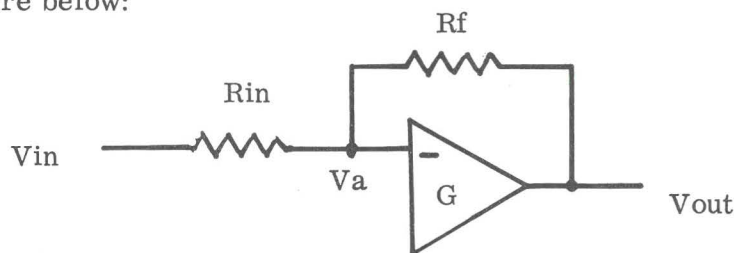
Consider the Amplifier shown below:



The amplifier is a Difference Amplifier with a positive and negative input and a gain of G . For the inputs A and B , the output Z is given by:

$$Z = G B - G A = G (B - A)$$

Consider the amplifier in a circuit with an input voltage V_{in} volts and an output voltage of V_{out} and with feedback and input resistors R_f and R_{in} as shown in the figure below:



Define the input voltage to the amplifier as V_a . By simple series circuit calculations using Ohm's Law, the voltage V_a is:

$$V_a = V_{out} (R_{in} / (R_{in} + R_f)) + V_{in} (R_f / (R_{in} + R_f))$$

Due to the feedback and amplifier gain, $V_{out} = - G V_a$.

Combining both equations and making the assumption that the gain is high compared to 1, the equation for the output voltage becomes:

$$V_{out} = - \frac{R_f}{R_{in}} V_{in}$$

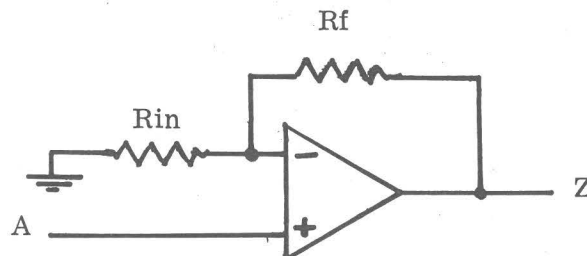
Fig. 63-1 shows various configurations and gains using different input and feedback elements. When a variable resistor (potentiometer X or Y) is used as the input or output element, the function performed is a division or multiplication by that resistance (or potentiometer shaft position).

PROCEDURE:

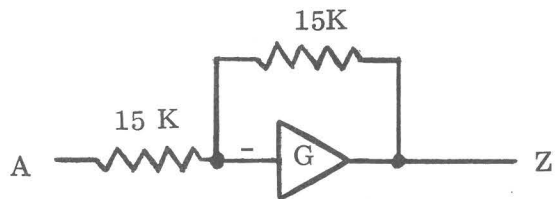
1. Wire the experiment in accordance with Fig. 63-2. The Threshold Logic element resistors are used as the input and feedback resistors. The feedback and input leads are connected to the selected resistors. The two Solid State/ LSI Logic resistors (3k) may also be used. They are used in Fig. 63-2 to provide a reference or mid voltage to the + input of the Operational Amplifier.
2. Select an R_f and R_{in} to give a gain greater than 1 . Calculate the gain.
$$\text{Gain} =$$
3. Vary the input (potentiometer X) and record the input voltage V_{in} and the output voltage V_{out} on a data sheet. Use over ten different points. Use Table A
4. Plot the results. Determine the gain as the " change in output " divided by the "change in input" over a segment of the curve. Compare the calculated and measured gain and calculate the percentage difference. Use Graph A
5. Select another combination of R_f and R_{in} for a gain less than 1 (i.e. , attenuation) . Vary the inputs and record the input and output. Plot the results and calculate the gain of the system. Use Table B and Graph B.
6. Use Potentiometer Y for R_f and all four resistors in parallel as shown in Fig. 63-3 for R_{in} . Measure the gain using only two points. Plot the gain verses R_f for positions of Pot Y on markers 0, 1, 2, 3, etc. to 10. This system is then a variable gain amplifier. What gain markings would one place on each of the potentiometer markings ? Use Table C and Graph C.
7. Use potentiometer Y for R_{in} and the parallel set of resistors for R_f . Again measure the gain for various positions of Y using two input output readings. Plot the results of gain verses the position of Y . Comment on the difference between this and the previous test. What is the range of gains achievable ? How can this be used for division ? Use Table D and Graph D.
8. Other configurations that can be studied are the Integrator, Rectifier, and Positive Gain circuits. These are shown in Fig. 63-4 . Select any number of these circuits and plot the input-output characteristics. Note that the rectifier has a non-linear output. The output of the integrator circuit goes positive or negative depending on X . The rate is determined by the displacement or rotation of potentiometer X from its center position. Potentiometer Y can be used as the + input in either the Integrator or the Rectifier to determine the position of X for zero rate, or the exact voltage for the start of the rectifier output.

EXERCISES:

1. Select resistors and draw the circuits for Operational Amplifier configurations with gains of : (a) - 2 , (b) - 10 , (c) + 3 .
2. What is the equation for the output voltage for the following Op Amp configuration ? Try the experiment to check - out the equation .

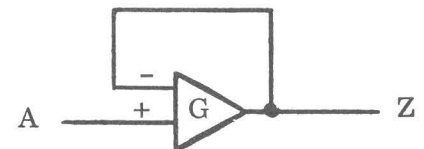


3. What happens when the diodes are reversed in Fig. 63-4 ?



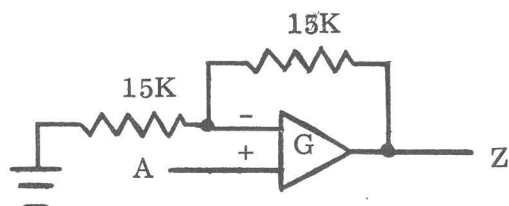
$$Z = -A$$

INVERSION



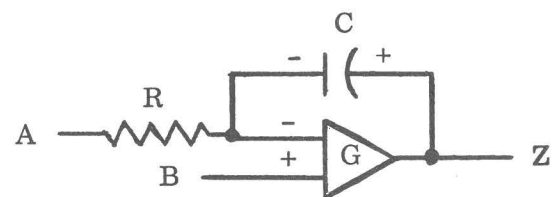
$$Z = +A$$

FOLLOWING



$$Z = +2A$$

GAIN (POSITIVE)



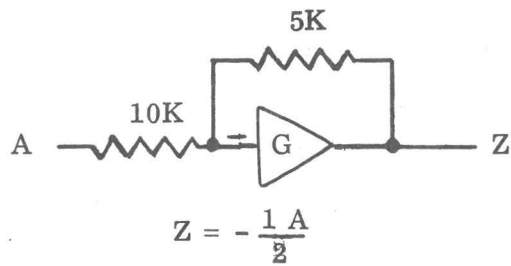
$$Z \propto \int (A - B) dt$$

INTEGRATION

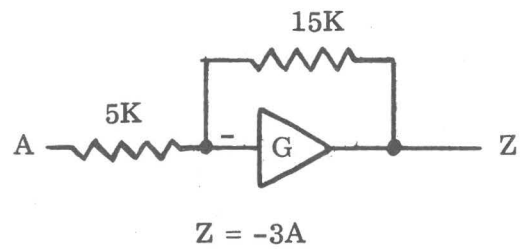
Figure 63 - 1 OP. AMP. CONFIGURATIONS

Operational Amplifiers

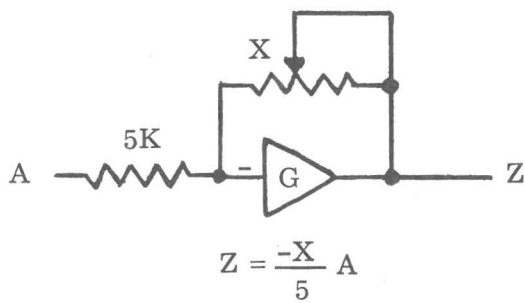
Experiment No. 63 (page 3)



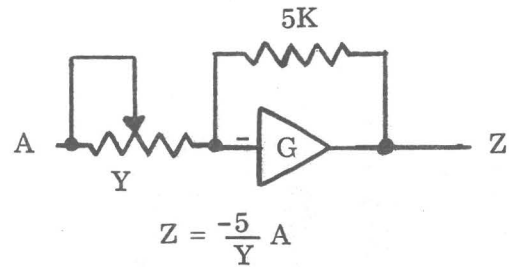
ATTENUATION



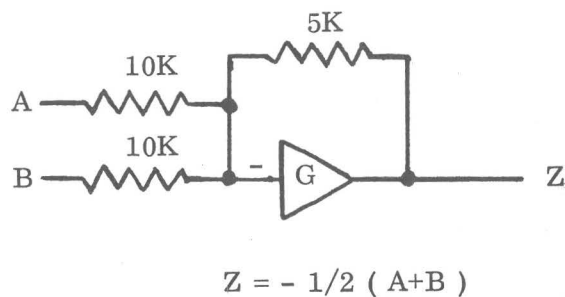
GAIN (NEGATIVE)



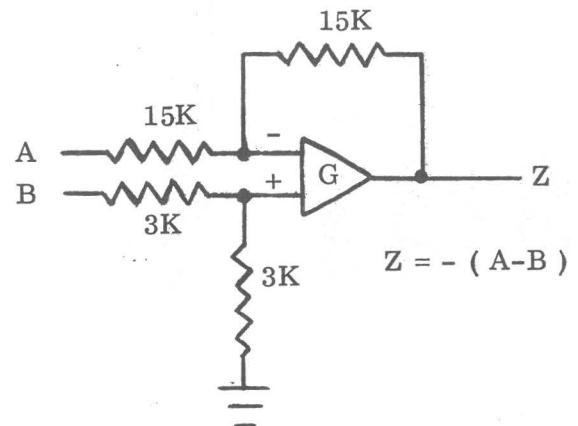
MULTIPLICATION



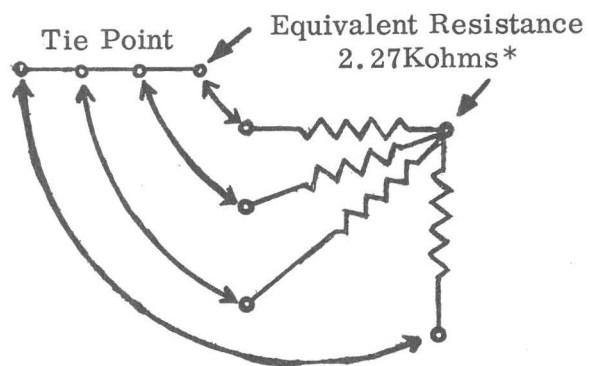
DIVISION



ADDING



SUBTRACTING



OP. AMP. GAIN

TABLE A

GRAPH A

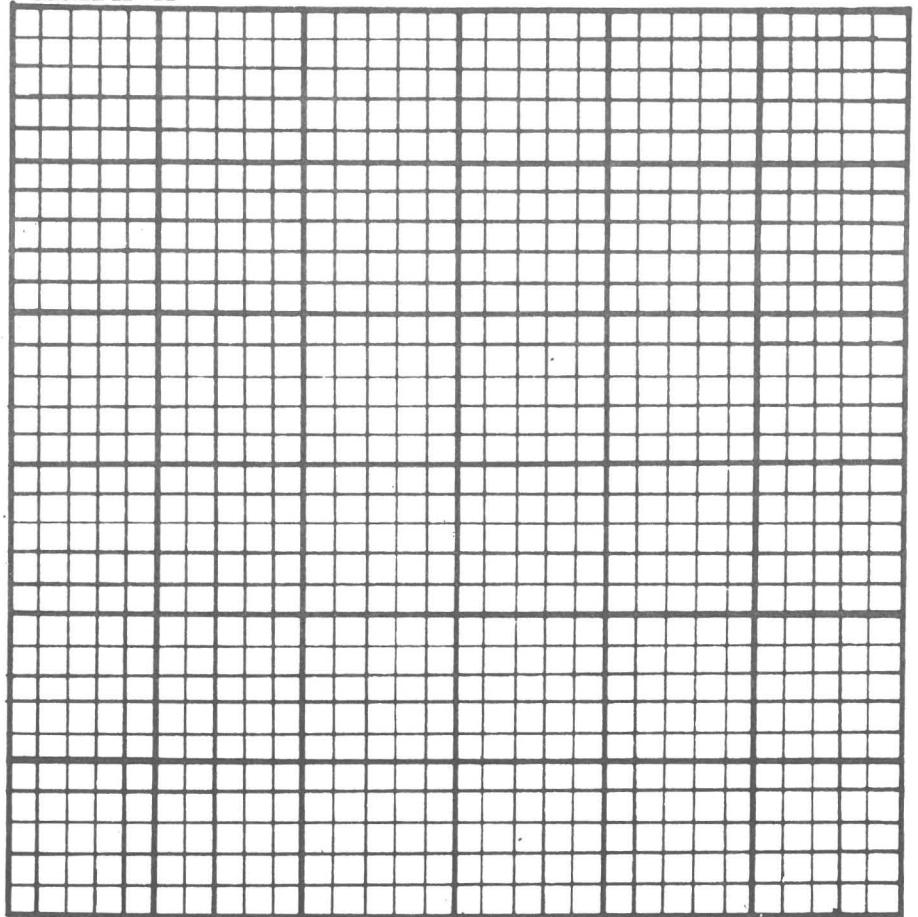


TABLE B

GRAPH B

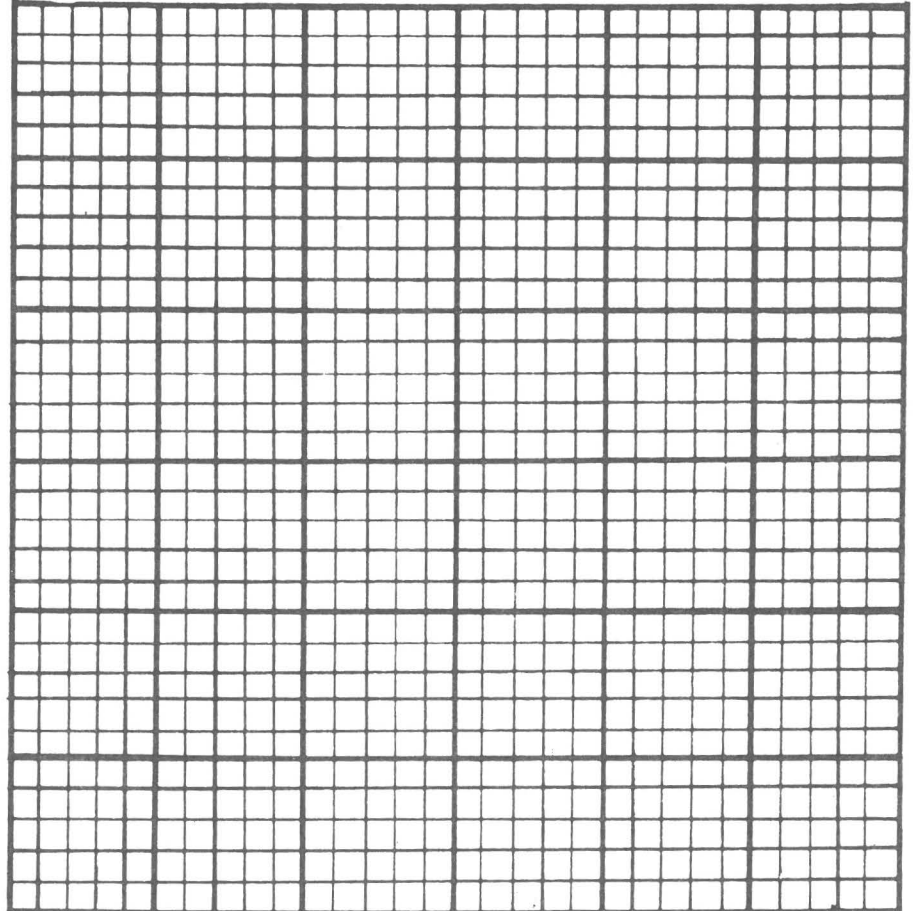


TABLE C

GRAPH C

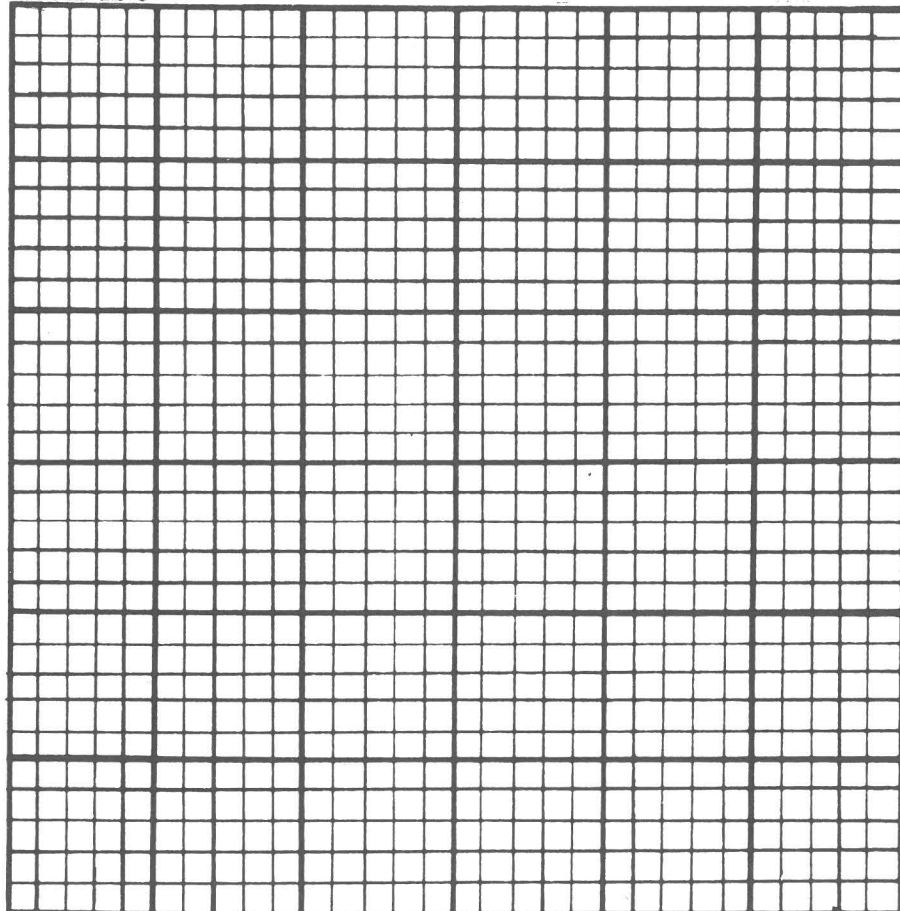
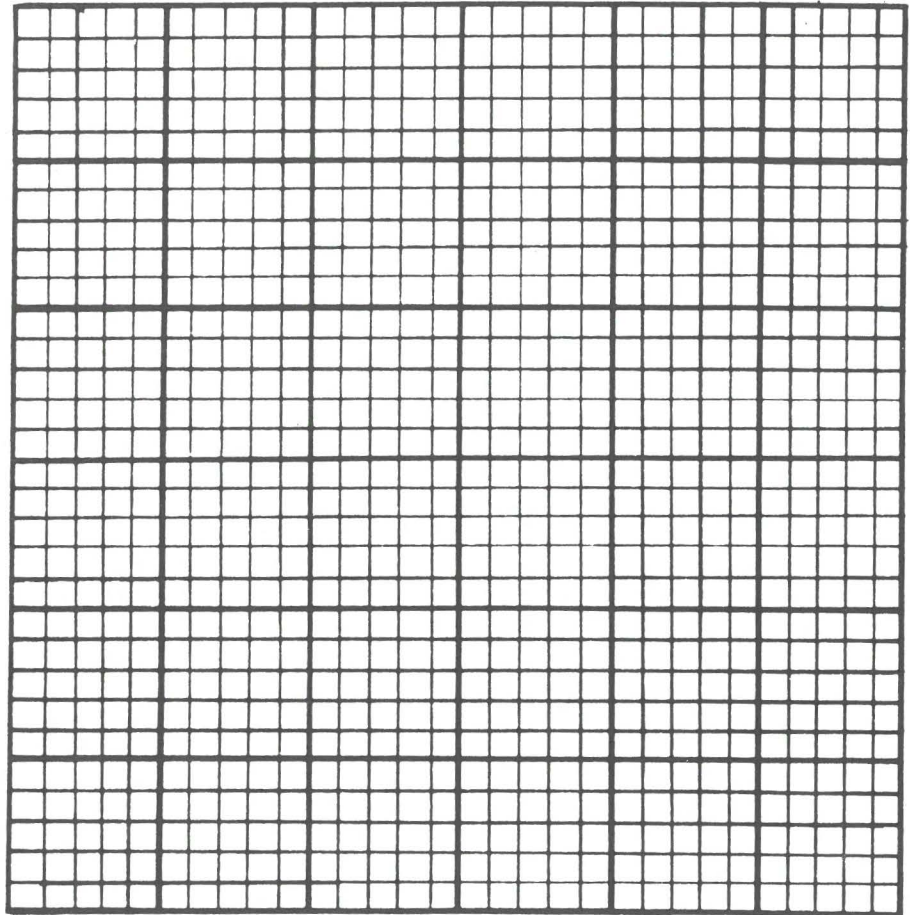


TABLE D

GRAPH D



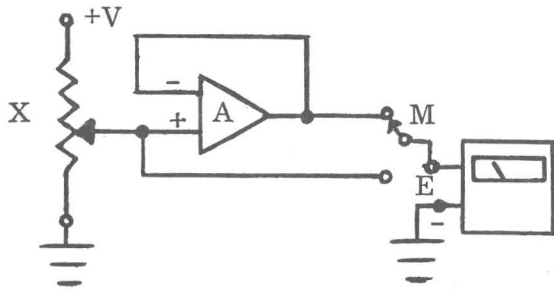


Fig. 64 - 4 (a) FOLLOWER

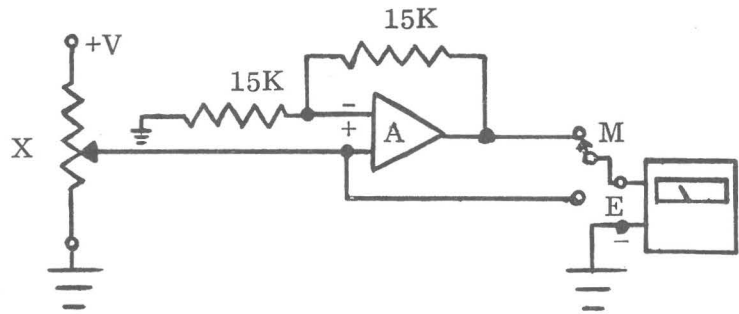


Fig. 63 - 4 (b) POSITIVE GAIN

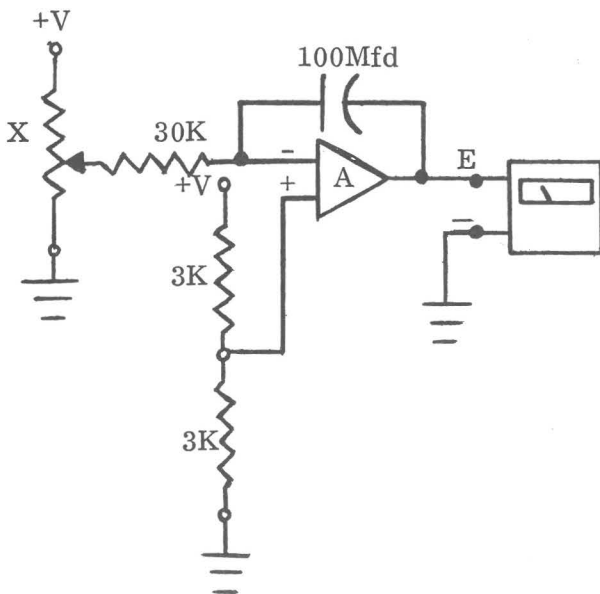


Fig. 63 - 4 (c) INTEGRATOR

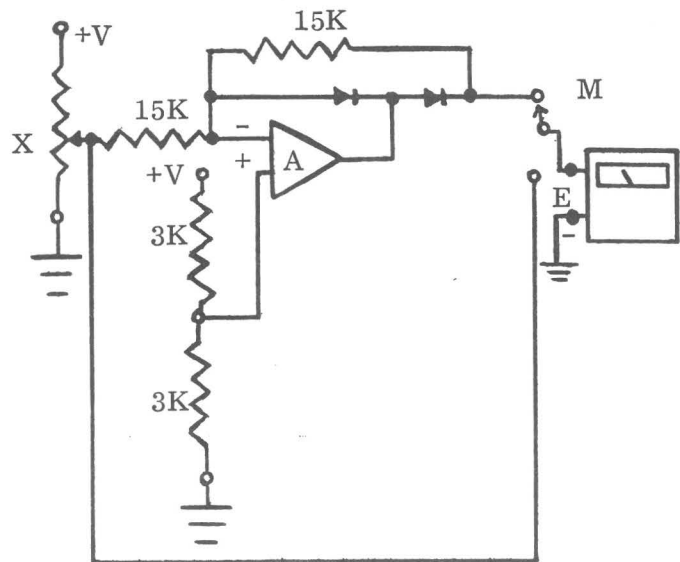


Fig. 63 - 4 (d) LINEAR RECTIFIER

NOTE:

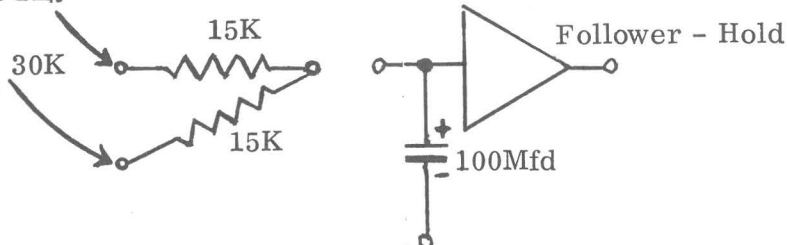


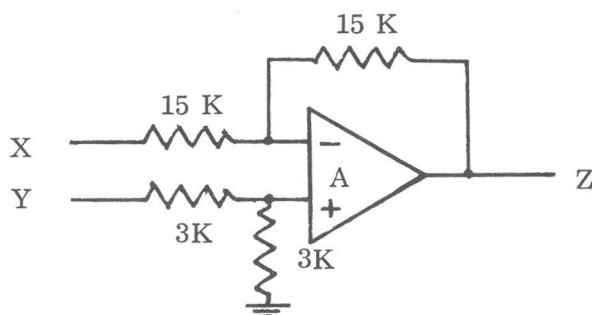
Fig. 63-4(e) FOLLOWER-HOLD

OBJECTIVE:

The previous experiment described the Operational Amplifier and various applications in analog signal processing and computing. The Op Amp has two inputs for + and - effects on the output. These are used to supply reference voltages in the previous experiments. In many applications it is necessary to measure the difference between two voltages. This experiment illustrates the use of the Op Amp as a Difference Amplifier. The characteristics of the Op Amp are also measured. The actual Gain G is first measured. This is the difference gain of the amplifier. The gain when there is a zero difference input is defined as the "common mode gain" and is also measured. A performance factor for the Op Amp is called the "common mode rejection ratio" and is also determined.

DESCRIPTION:

The Op Amp configuration shown below generates an output corresponding to the difference between the two inputs X and Y .



There are generally small voltage "offsets" and "errors" in the output depending on the performance of the Operational Amplifier. Consider a set of inputs $X = 2$ and $Y = 2.5$. For another set of inputs $X = 2.3$ and $Y = 2.8$, the difference between X and Y are the same, i.e., 0.5 . The increase in voltage from 2 to 2.3 is 0.3 and is "common" to both X and Y . Such a common increase is referred to as "common mode". The output for a common difference should naturally be independent of the input levels. They are not, and the extent to which the output does remain the same is a "figure of merit" of the Op Amp and is called the "common mode rejection ratio". The normal gain of the amplifier is the change of output divided by the change of input. This is also called the "difference gain". The common mode gain is defined as the change in output for a level shift in both inputs. The "common mode rejection ratio" is the ratio of the Difference Gain to the Common Mode Gain. It is desirable to have a high difference gain and to have a low common mode gain.

The first part of this experiment will demonstrate the Op Amp used to measure differences with feedback to provide an overall gain of 1 . The second part will measure the difference gain or the "open loop" gain without feedback and the common mode gain to determine the CMRR (common mode rejection ratio).

PROCEDURE:

1. Wire the Circuit shown in Fig. 64-1. The loose lead from the meter +E terminal is touched to X and Y (the inputs) and to Z (the output) to read the voltages.

2. Set in various values of X and Y and read the output Z . To make sure that the amplifier operates in the linear region, make sure that the output Z stays within 0.5 to 4.5 volts. Enter the data in Table A. When complete, calculate $(Y - X)$ and tabulate in another column. Plot $(Y - X)$ horizontally and Z vertically in Graph A.

3. Wire the Circuit in Fig. 64-2 to measure the Difference Gain. A constant voltage input is placed on the + input. The meter terminals marked +E and - are used to measure the output voltage with - connected to Ground and + E connected to the output. The input voltage difference is so small (for large gains) that one cannot measure the input voltages to + and - and subtract the difference. It is necessary to measure the difference directly. The meter terminals marked + I and - are used. Using this scale, the full scale meter reading is 0.05 volts or 50 millivolts. It is necessary to change the lead connections for inputs and outputs. When reading the input, if the meter reads below zero, change the polarity of the connection to the meter. When reading the input, be sure to record the polarity of input readings (i.e., is the +I meter input connected to the + or the - input to the Op Amp) .

4. Enter all Data taken into TABLE B . Vary X for different outputs ranging from 0.5 to 4.5 volts. Take about 7 different points. Plot the outputs on a graph. B. Plot Output vertically and input horizontally. Determine the slope of the curve and the gain.

Difference Gain = _____

5. Wire the Circuit shown in Fig. 64-3 to determine the common mode gain. The X input is applied to both the + and - Op Amp inputs. Use the meter to read the output voltages for various different input voltages . The output should vary between 0.5 and 4.5 volts with about 4 different readings. Plot Vout vs X . Determine the slope or the Common Mode Gain (change in Vout divided by change in X). Use table and graph C.

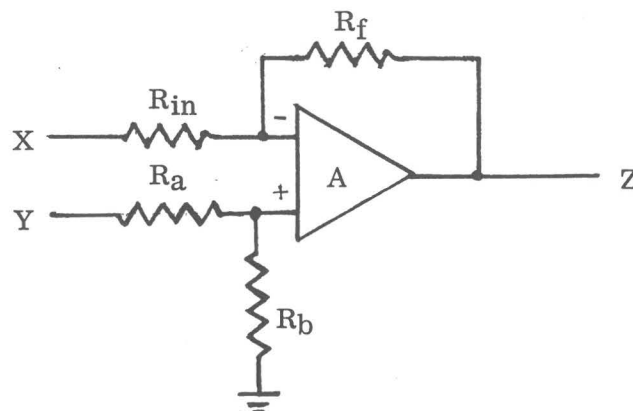
Common Mode Gain = _____

6. Calculate the common mode rejection ratio.

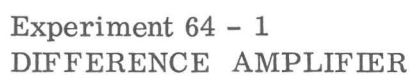
$$\text{CMRR} = \frac{\text{Difference Gain}}{\text{Common Mode Gain}} = \frac{\quad}{\quad} = \quad$$

QUESTIONS:

1. Determine the Transfer Function (Output verses input equation) for the following Op Amp Difference Circuit.

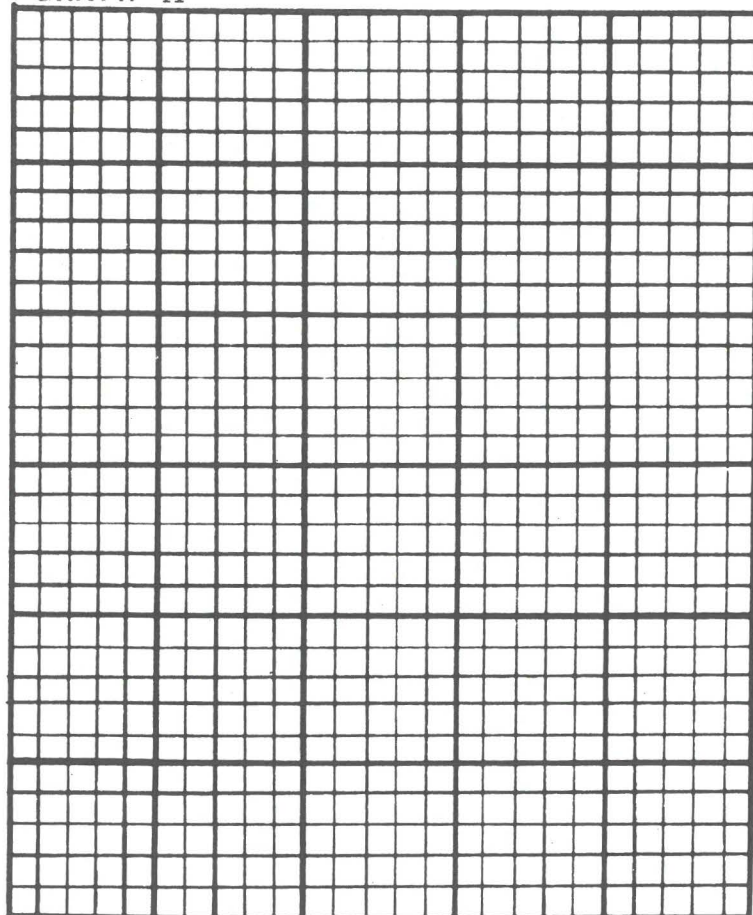


Experiment No. 64 (page 3)

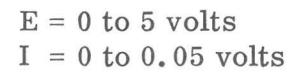


INPUTS	OUTPUTS
1. <i>Information</i>	1. <i>Information</i>
2. <i>Knowledge</i>	2. <i>Knowledge</i>
3. <i>Skills</i>	3. <i>Skills</i>
4. <i>Attitudes</i>	4. <i>Attitudes</i>
5. <i>Values</i>	5. <i>Values</i>
6. <i>Beliefs</i>	6. <i>Beliefs</i>
7. <i>Emotions</i>	7. <i>Emotions</i>
8. <i>Behaviors</i>	8. <i>Behaviors</i>
9. <i>Relationships</i>	9. <i>Relationships</i>
10. <i>Environment</i>	10. <i>Environment</i>
11. <i>Resources</i>	11. <i>Resources</i>
12. <i>Time</i>	12. <i>Time</i>
13. <i>Energy</i>	13. <i>Energy</i>
14. <i>Health</i>	14. <i>Health</i>
15. <i>Finances</i>	15. <i>Finances</i>
16. <i>Education</i>	16. <i>Education</i>
17. <i>Work</i>	17. <i>Work</i>
18. <i>Family</i>	18. <i>Family</i>
19. <i>Community</i>	19. <i>Community</i>
20. <i>Society</i>	20. <i>Society</i>
21. <i>World</i>	21. <i>World</i>
22. <i>Universe</i>	22. <i>Universe</i>
23. <i>God</i>	23. <i>God</i>
24. <i>Heaven</i>	24. <i>Heaven</i>
25. <i>Hell</i>	25. <i>Hell</i>
26. <i>Angels</i>	26. <i>Angels</i>
27. <i>Devils</i>	27. <i>Devils</i>
28. <i>Spirits</i>	28. <i>Spirits</i>
29. <i>Demons</i>	29. <i>Demons</i>
30. <i>Witches</i>	30. <i>Witches</i>
31. <i>Magicians</i>	31. <i>Magicians</i>
32. <i>Warlocks</i>	32. <i>Warlocks</i>
33. <i>Wizards</i>	33. <i>Wizards</i>
34. <i>Enchanters</i>	34. <i>Enchanters</i>
35. <i>Summoners</i>	35. <i>Summoners</i>
36. <i>Conjurers</i>	36. <i>Conjurers</i>
37. <i>Illusionists</i>	37. <i>Illusionists</i>
38. <i>Transmuters</i>	38. <i>Transmuters</i>
39. <i>Alchemists</i>	39. <i>Alchemists</i>
40. <i>Blacksmiths</i>	40. <i>Blacksmiths</i>
41. <i>Smiths</i>	41. <i>Smiths</i>
42. <i>Blacksmiths</i>	42. <i>Blacksmiths</i>
43. <i>Smiths</i>	43. <i>Smiths</i>
44. <i>Blacksmiths</i>	44. <i>Blacksmiths</i>
45. <i>Smiths</i>	45. <i>Smiths</i>
46. <i>Blacksmiths</i>	46. <i>Blacksmiths</i>
47. <i>Smiths</i>	47. <i>Smiths</i>
48. <i>Blacksmiths</i>	48. <i>Blacksmiths</i>
49. <i>Smiths</i>	49. <i>Smiths</i>
50. <i>Blacksmiths</i>	50. <i>Blacksmiths</i>
51. <i>Smiths</i>	51. <i>Smiths</i>
52. <i>Blacksmiths</i>	52. <i>Blacksmiths</i>
53. <i>Smiths</i>	53. <i>Smiths</i>
54. <i>Blacksmiths</i>	54. <i>Blacksmiths</i>
55. <i>Smiths</i>	55. <i>Smiths</i>
56. <i>Blacksmiths</i>	56. <i>Blacksmiths</i>
57. <i>Smiths</i>	57. <i>Smiths</i>
58. <i>Blacksmiths</i>	58. <i>Blacksmiths</i>
59. <i>Smiths</i>	59. <i>Smiths</i>
60. <i>Blacksmiths</i>	60. <i>Blacksmiths</i>
61. <i>Smiths</i>	61. <i>Smiths</i>
62. <i>Blacksmiths</i>	62. <i>Blacksmiths</i>
63. <i>Smiths</i>	63. <i>Smiths</i>
64. <i>Blacksmiths</i>	64. <i>Blacksmiths</i>
65. <i>Smiths</i>	65. <i>Smiths</i>
66. <i>Blacksmiths</i>	66. <i>Blacksmiths</i>
67. <i>Smiths</i>	67. <i>Smiths</i>
68. <i>Blacksmiths</i>	68. <i>Blacksmiths</i>
69. <i>Smiths</i>	69. <i>Smiths</i>
70. <i>Blacksmiths</i>	70. <i>Blacksmiths</i>
71. <i>Smiths</i>	71. <i>Smiths</i>
72. <i>Blacksmiths</i>	72. <i>Blacksmiths</i>
73. <i>Smiths</i>	73. <i>Smiths</i>
74. <i>Blacksmiths</i>	74. <i>Blacksmiths</i>
75. <i>Smiths</i>	75. <i>Smiths</i>
76. <i>Blacksmiths</i>	76. <i>Blacksmiths</i>
77. <i>Smiths</i>	77. <i>Smiths</i>
78. <i>Blacksmiths</i>	78. <i>Blacksmiths</i>
79. <i>Smiths</i>	79. <i>Smiths</i>
80. <i>Blacksmiths</i>	80. <i>Blacksmiths</i>
81. <i>Smiths</i>	81. <i>Smiths</i>
82. <i>Blacksmiths</i>	82. <i>Blacksmiths</i>
83. <i>Smiths</i>	83. <i>Smiths</i>
84. <i>Blacksmiths</i>	84. <i>Blacksmiths</i>
85. <i>Smiths</i>	85. <i>Smiths</i>
86. <i>Blacksmiths</i>	86. <i>Blacksmiths</i>
87. <i>Smiths</i>	87. <i>Smiths</i>
88. <i>Blacksmiths</i>	88. <i>Blacksmiths</i>
89. <i>Smiths</i>	89. <i>Smiths</i>
90. <i>Blacksmiths</i>	90. <i>Blacksmiths</i>
91. <i>Smiths</i>	91. <i>Smiths</i>
92. <i>Blacksmiths</i>	92. <i>Blacksmiths</i>
93. <i>Smiths</i>	93. <i>Smiths</i>
94. <i>Blacksmiths</i>	94. <i>Blacksmiths</i>
95. <i>Smiths</i>	95. <i>Smiths</i>
96. <i>Blacksmiths</i>	96. <i>Blacksmiths</i>
97. <i>Smiths</i>	97. <i>Smiths</i>
98. <i>Blacksmiths</i>	98. <i>Blacksmiths</i>
99. <i>Smiths</i>	99. <i>Smiths</i>
100. <i>Blacksmiths</i>	100. <i>Blacksmiths</i>

GRAPH A

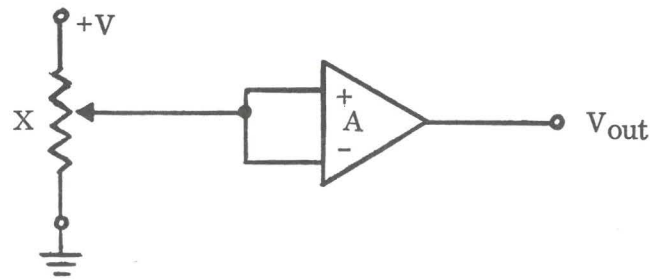


Experiment No. 64 (page 4)



Experiment 64 - 2
DIFFERENCE GAIN

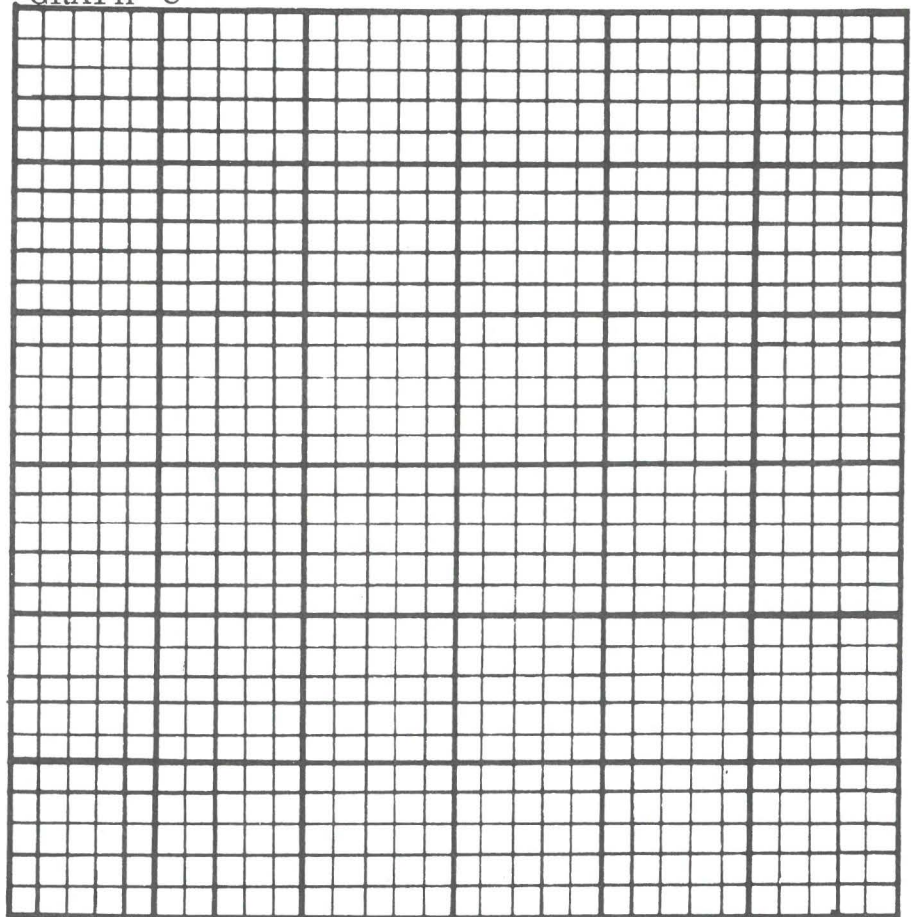
[illegible][illegible]



Experiment 64 - 3
COMMON MODE GAIN

TABLE C

GRAPH C

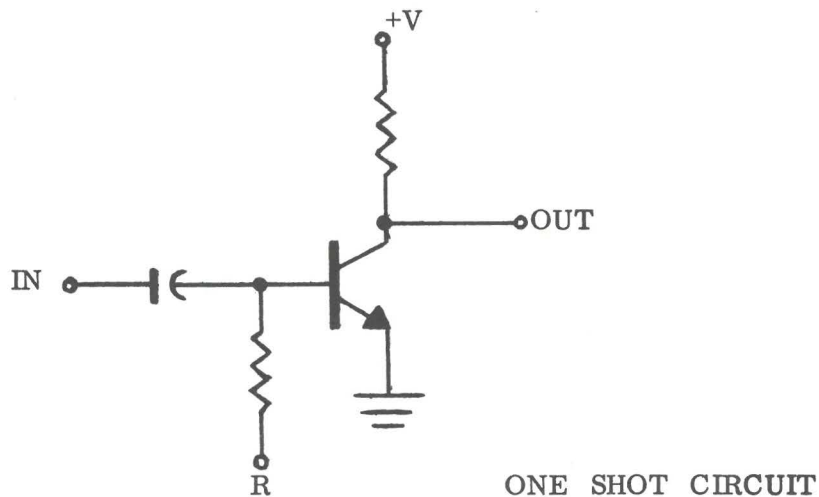


OBJECTIVE:

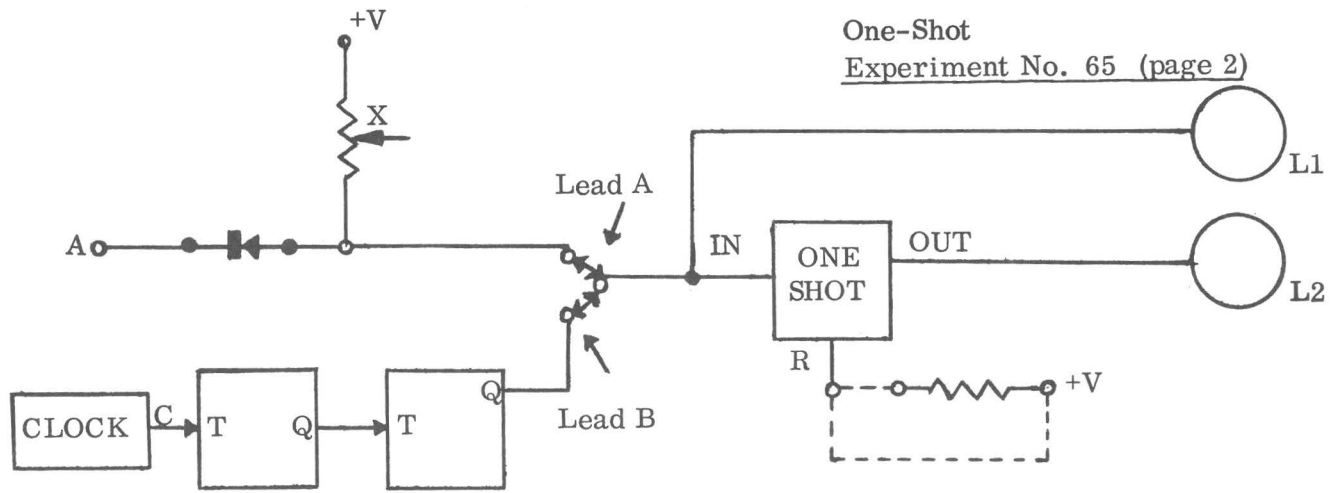
This experiment demonstrates a "one-shot" or single pulse circuit operation. The circuit element can either generate a short positive pulse or a short negative pulse.

DESCRIPTION(Lagging Edge One Shot):

When the one-shot "Return" (R) lead is connected to plus (+V) volts (1), the circuit transistor is turned ON and the output is "0". A "negative-going" input results in the transistor being turned off momentarily causing a positive pulse or a short momentary "1". The input can be a switch (A) or the clock.

PROCEDURE:

Wire the logic. (1) With lead A connected the one-shot is controlled by switch A. Slowly activate switch A (up and down). Observe the output. (2) With lead B connected, the one-shot is controlled by a 4 second pulse. (3) With a resistor ($R_1 + R_2$ or Threshold Resistors) connected to R and "1" the pulse is extended. Observe L1 and L2 and record the output in the data table for all conditions.



Experiment Logic 65-1
LAGGING EDGE ONE - SHOT

Record Data Below :

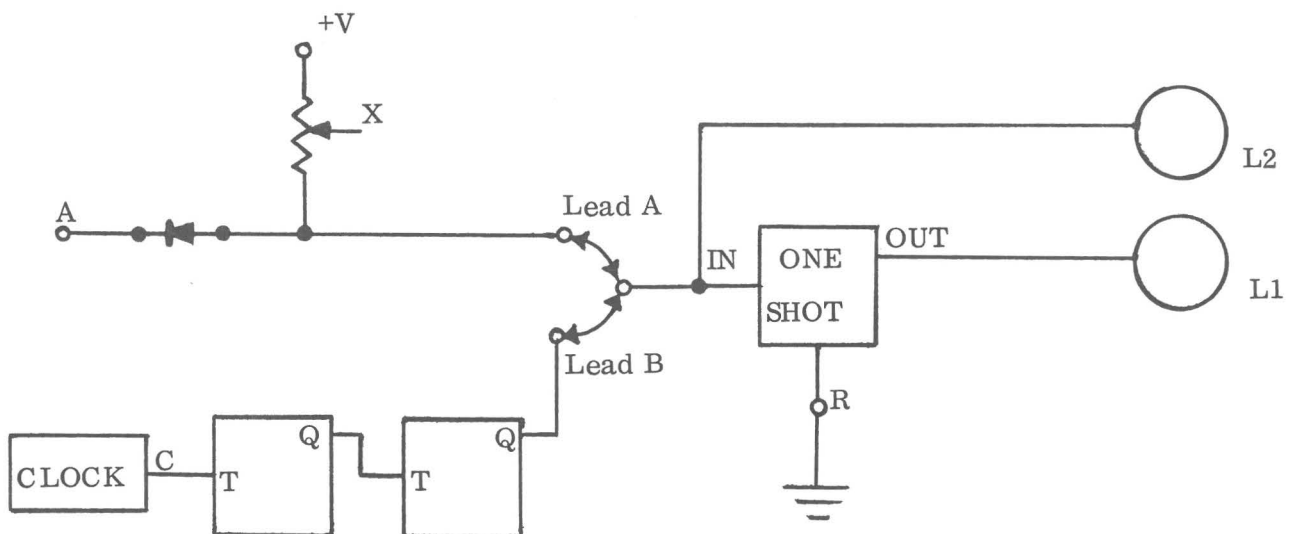
ONE - SHOT "R"	RESULTS - PULSE WIDTH
(1) Direct to +V	
(2) Via R1 (3K)	
(3) Via Threshold R = 5K	
(4) R = 15K	
(5) R = 30K	
(6)	
(7)	
(8)	

QUESTIONS:

1. Does L1 go ON when the input L2 goes ON (1) or OFF (0)?
2. What is the time duration of the output one-shot pulse?
3. Is the lagging-edge one-shot output normally a 1 or a 0?
4. Draw the logic to provide a lagging-edge one-shot with a normally ON (1) lamp and a one-shot OFF pulse.
5. Draw the logic for a lagging-edge one-shot to generate a short positive pulse triggered by a positive going input pulse.

DESCRIPTION (Leading-Edge One-Shot):

When the One-Shot return (R) is connected to the ground (0), the transistor is kept off, and the one-shot output is "1". When a positive going input (0 to 1) is applied to the input (capacitor), the transistor is momentarily turned ON, and the output drops to "0". After the input capacitor charges, the output returns to a "1" resulting in a short "0" pulse.



Experiment Logic 65-2

LEADING EDGE ONE - SHOT

PROCEDURE:

Wire the logic. (1) With lead A connected, the one-shot is controlled by Switch A. Activate the switch UP and DOWN slowly. (2) With Lead B connected, the one-shot is triggered by the 4-second clock. It is not possible to extend this pulse. Record all observations below:

SWITCH A	
CLOCK	

QUESTIONS:

5. When does L1 go ON?
6. What is the time duration of the pulse?
7. Draw the logic diagram for a leading edge one-shot with a short "1" pulse output.
8. Draw the diagram for a negative "0" pulse which is triggered by a negative input and where the length of the negative pulse may be varied.

OBJECTIVE:

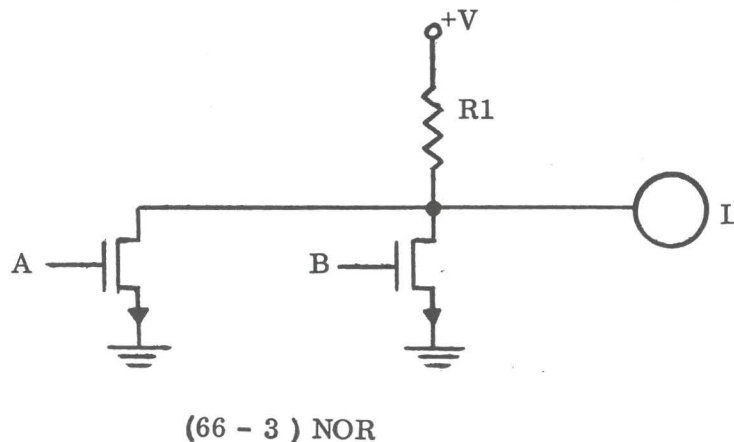
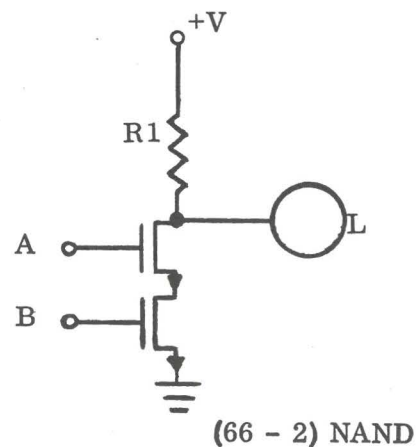
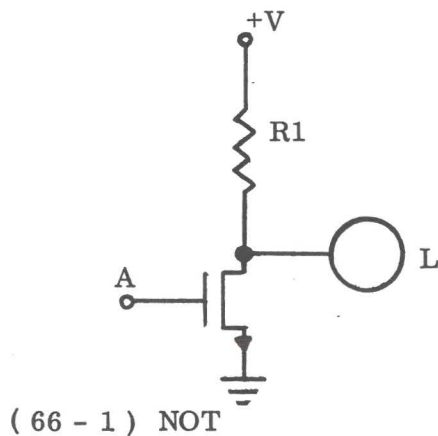
Large Scale Integration (LSI) systems and Medium Scale Integration (MSI) systems, contain semi-conductor switches in various combinations to yield different and complex logic functions. Hundreds of logic switches and logic functions are fabricated on a single chip only a fraction of an inch in size.

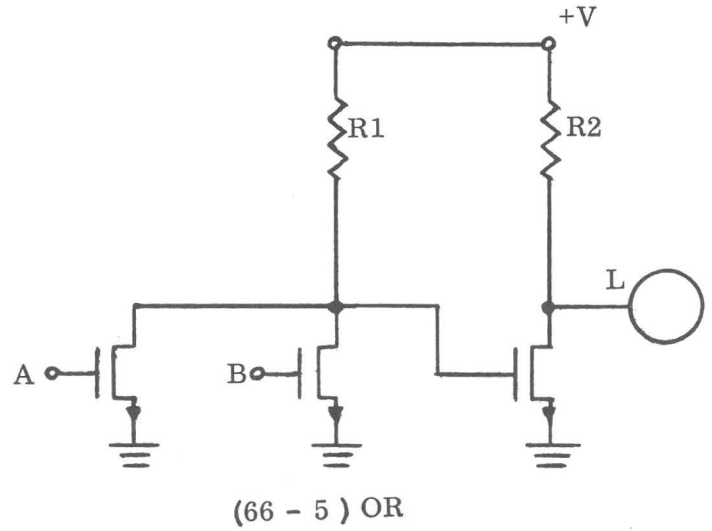
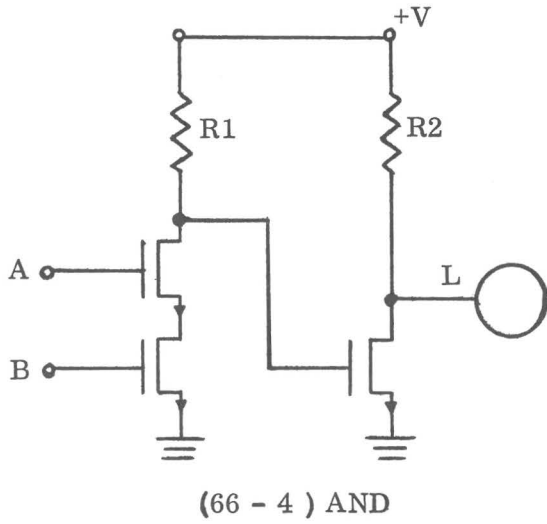
DESCRIPTION:

The semiconductor switch input is controlled by an input switch or the output of another gate. Each output must be used with a load (resistor). When discrete transistor switches are used, this type of logic is called TRANSISTOR LOGIC. All Logic element configurations can be implemented.

PROCEDURE:

Wire each logic shown on the following page. Use switches as the inputs. Set the inputs to the conditions given in the TRUTH TABLE. Check all outputs for the proper logic.





Enter Data Below :

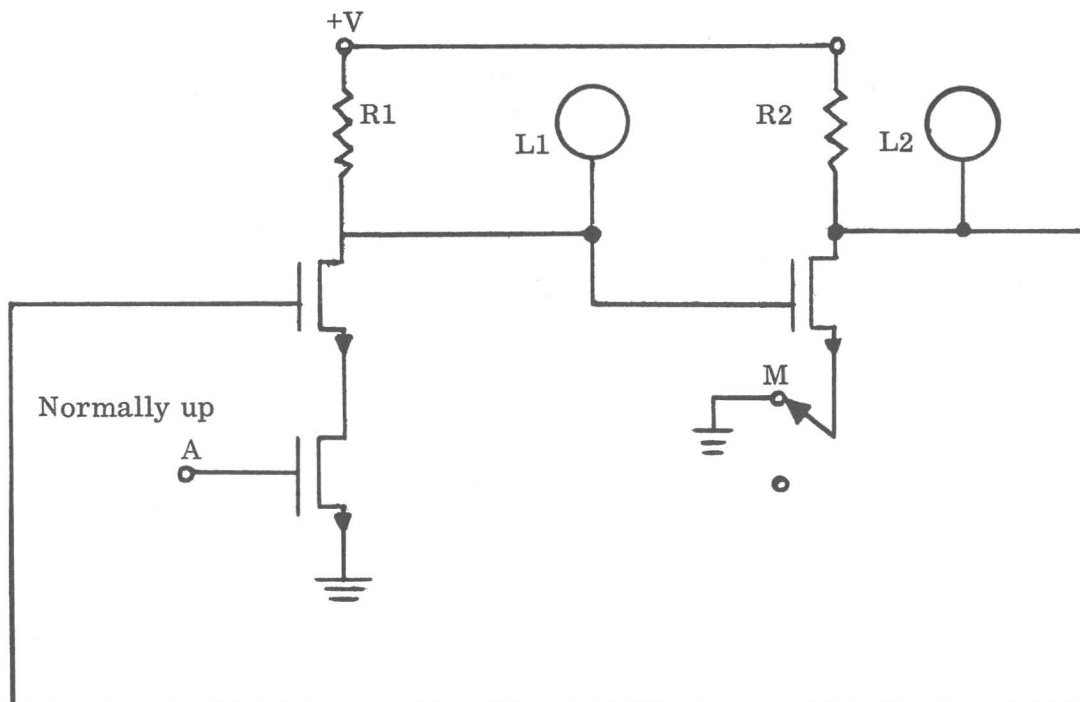
A	B	NOT	NOR	NAND	AND	OR
0	0					
0	1					
1	0					
1	1					

QUESTIONS:

1. Draw the LSI logic for $\bar{A} B$.
2. Draw the LSI logic for $\overline{ABC}$.
3. Draw the LSI logic for $A+B+C$.
4. Draw the LSI logic for $\bar{A}+B$.
5. Draw the LSI logic for a HALF ADDER with inputs A and B.
6. Draw the LSI logic for a FULL ADDER.

DESCRIPTION (Flip-Flop):

Two LSI switches are connected in end-around series to form a flip-flop memory in much the same manner as was used with the NOR or NAND logic.



(66 - 6) FLIP - FLOP

The flip-flop can be set to another state by an LSI switch in a series or in parallel with an output. Switch A is normally UP. When A is switched DOWN momentarily, L1 becomes a "1". When switch M is momentarily depressed, L2 becomes a "1".

PROCEDURE:

Wire the logic. Set switches A and M as shown in the table and record the data.

SWITCH	L1	L2
M		
A		
M		
M		
A		

QUESTIONS:

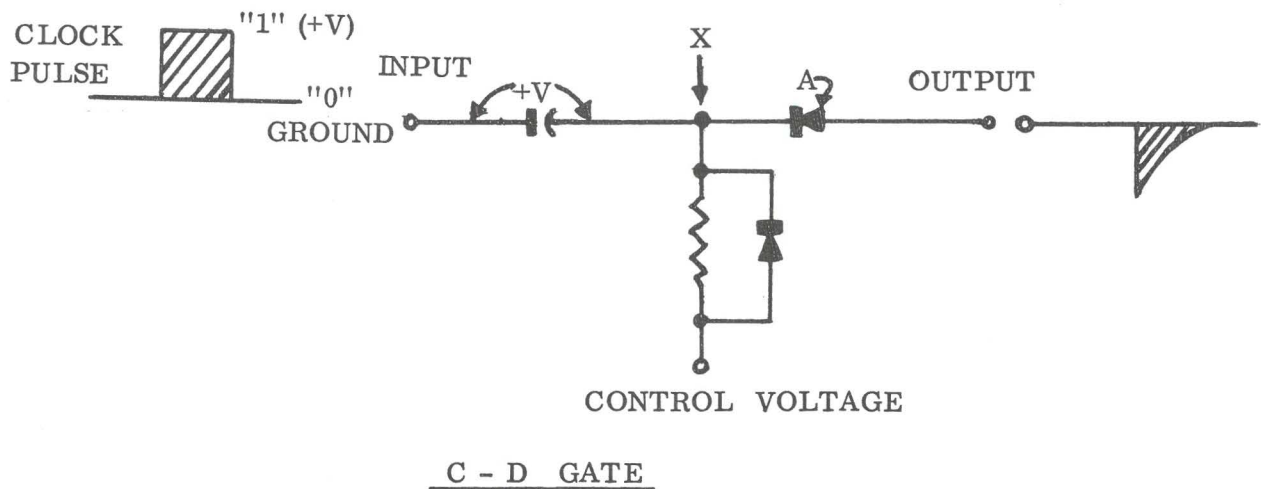
7. What occurs if both switches are used simultaneously?
8. Which switch is used to set L2 to a "1"?

OBJECTIVE:

This experiment demonstrates the operation of a Capacitor Diode gate (C-D gate). This gate is used to control the transmission of pulses. Control is effected by a voltage or a switch.

DESCRIPTION:

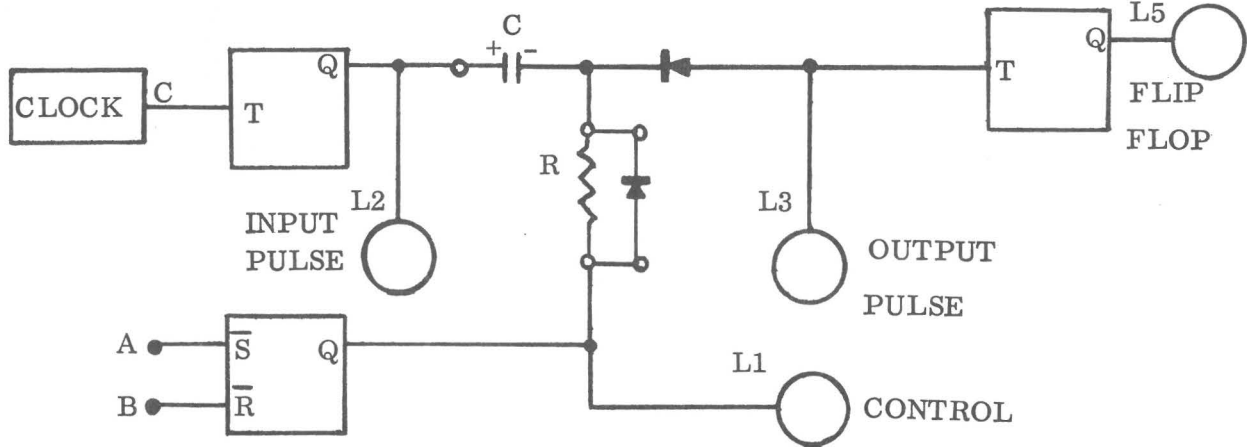
When the control voltage is "0" or ground the capacitor will charge to a "1" when the clock input voltage is high.



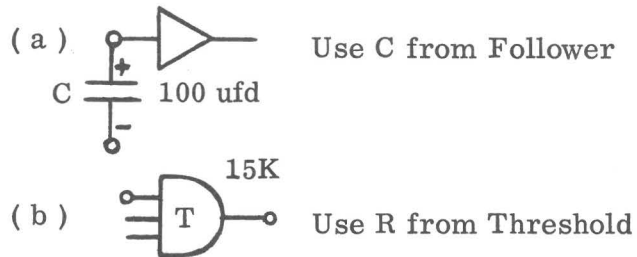
When the clock pulse drops to "0", the voltage at point "X" drops to a negative voltage, diode (A) is forward biased or on, and the negative pulse will be transmitted (to trigger a flip - flop). When the control voltage is +V or "1", the point X voltage will be at +V and the diode A is backbiased or OFF. A negative input or pulse step will not result in conduction of A and there will be no transmission of the pulse. This is then a two INPUT AND GATE with one control VOLTAGE input and pulse input with negative pulse edge transmission.

PROCEDURE:

Wire the logic. Use the HOLD Capacitor, the Resistors, and two DIODES as shown on the following page:



Experiment Logic 67-1
CAPACITOR DIODE GATE



Use switches A and B for the control and comment on the results.

QUESTIONS:

1. What is an advantage of the C-D gate over other logic gates?
2. Could a C-D gate be used with the diode in the opposite direction?
3. Can a C-D gate of this type be expected to drive any device requiring a positive current flow?

OBJECTIVE:

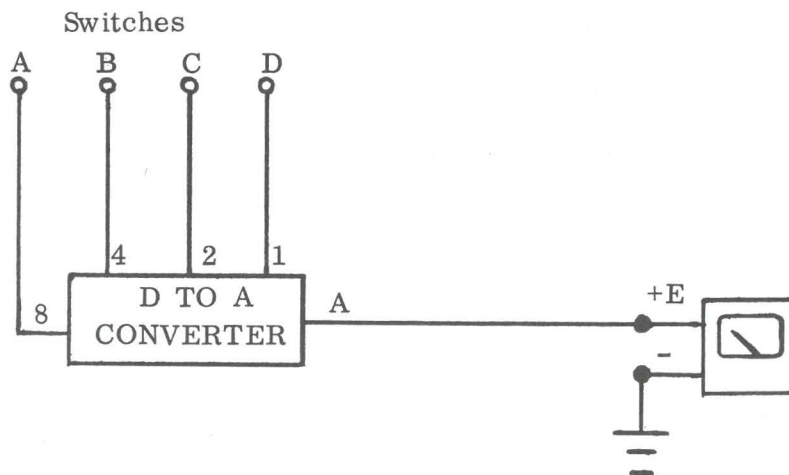
This experiment demonstrates the D to A converter circuit which converts binary inputs to an analog voltage.

DESCRIPTION (D to A Converter):

The inputs are provided via the binary switches and are coded in the regular binary manner (8, 4, 2, 1). The output voltage is read on the meter. The circuit operates with different "resistors" for each of the inputs, where the currents through the "resistors" (and into the meter) vary in proportion to 8, 4, 2, 1. The reading is then proportional to the binary number.

PROCEDURE:

Wire the experiment shown below:



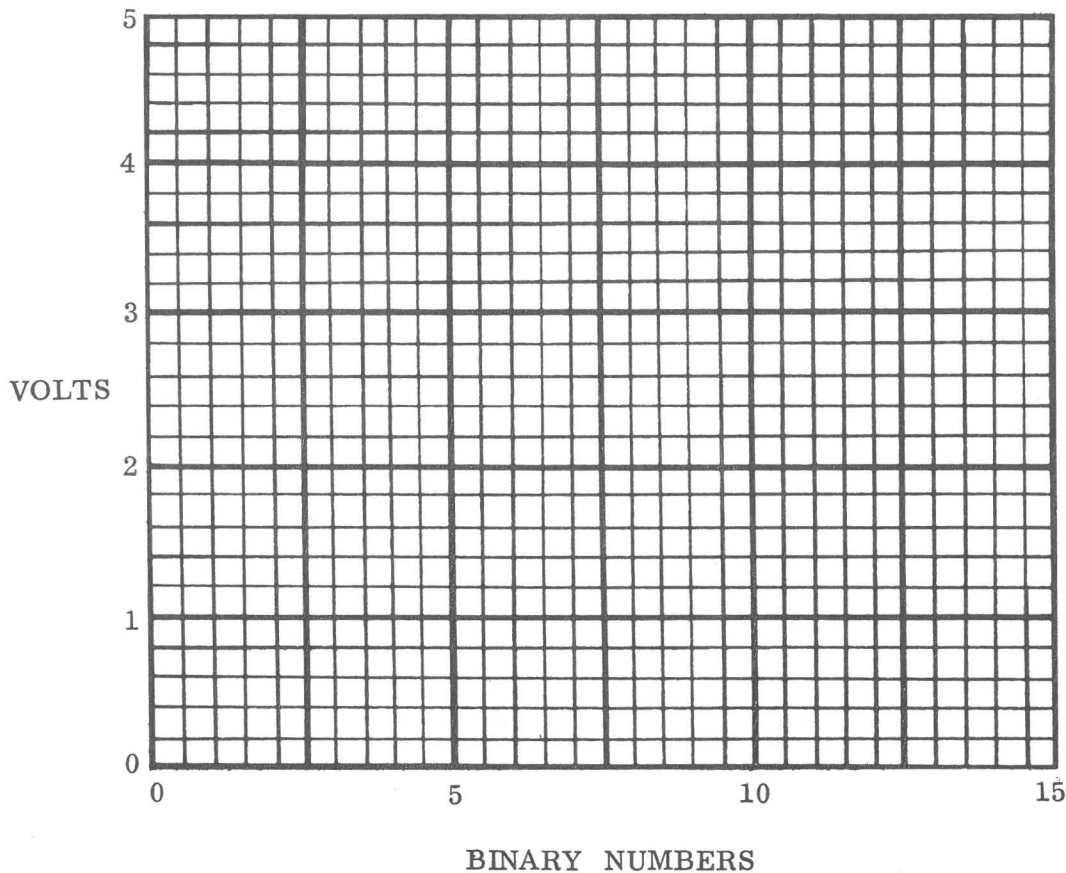
Experiment Logic 68-1

DIGITAL TO ANALOG CONVERTER

Record the binary number and the corresponding analog voltages in the table below:

A	B	C	D	#	Volts
0	0	0	0	0	
0	0	0	1	1	
0	0	1	0	2	
0	0	1	1	3	
0	1	0	0	4	
0	1	0	1	5	
0	1	1	0	6	
0	1	1	1	7	
1	0	0	0	8	
1	0	0	1	9	
1	0	1	0	10	
1	0	1	1	11	
1	1	0	0	12	
1	1	0	1	13	
1	1	1	0	14	
1	1	1	1	15	

Plot the results on a graph. Draw a "best" straight line through the points.



QUESTIONS:

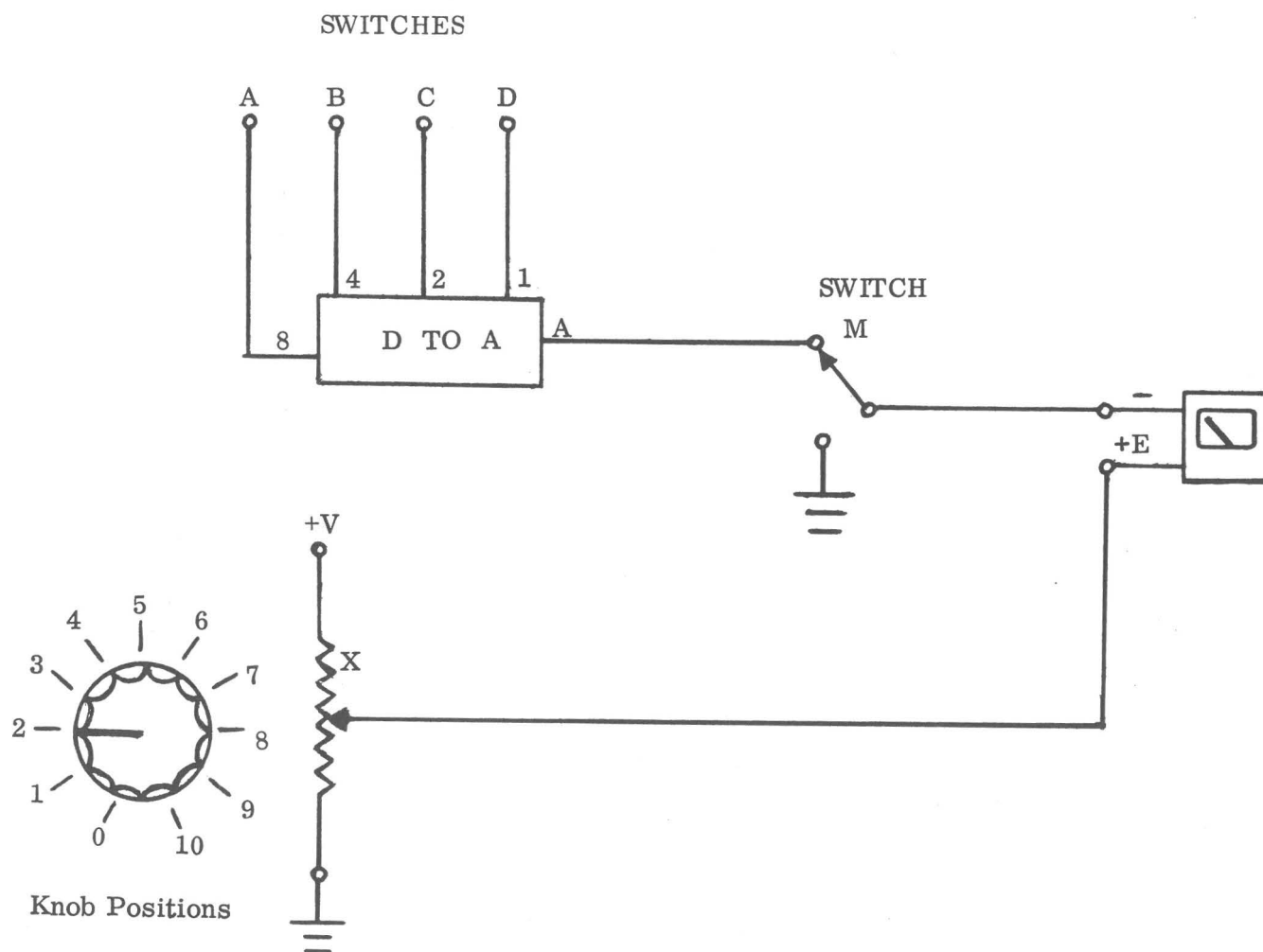
1. If each digital count represents two pounds of weight, how many volts would one count correspond to in the analog voltage ?
2. What is the analog scale factor in "pounds per volt" ?
3. What is the digital scale factor in "pounds per bit" ?
4. What is the scale factor in "volts per pound" ?
5. What is the "resolution" of the converter (given as a percentage of one count per full scale) ?
6. What is the offset (voltage reading for zero binary count) ?
7. What is the non-linearity (voltage of points off from "best fit" line expressed as a percentage of full scale) ?

OBJECTIVE:

This experiment demonstrates how an analog input voltage is applied to a Logic System which generates the equivalent binary number.

DESCRIPTION:

The D to A converter is used with a "feedback" or comparison method. The analog voltage is applied to a "zero or comparison or null" device such as the meter.



Experiment Logic 69 - 1

ANALOG TO DIGITAL CONVERSION

The potentiometer (X) is set at to an arbitrary voltage level. When switch M is DOWN, the meter reads this input voltage. When switch M is UP, the "D to A" analog output is connected to the (-) terminal of the meter. When the binary number results in an analog voltage equal to or greater than "X" volts, the meter will read 0 volts, or below zero.

PROCEDURE:

Wire the logic. Set X to some arbitrary voltage and read X. (Switch M down). With Switch M up, step the binary switches thru the binary numbers.(0000, 0001, etc., to 1111). Stop at the number which gives the closest positive reading to 0 volts. This is then the binary output corresponding to the analog voltage. Enter various readings in the table.

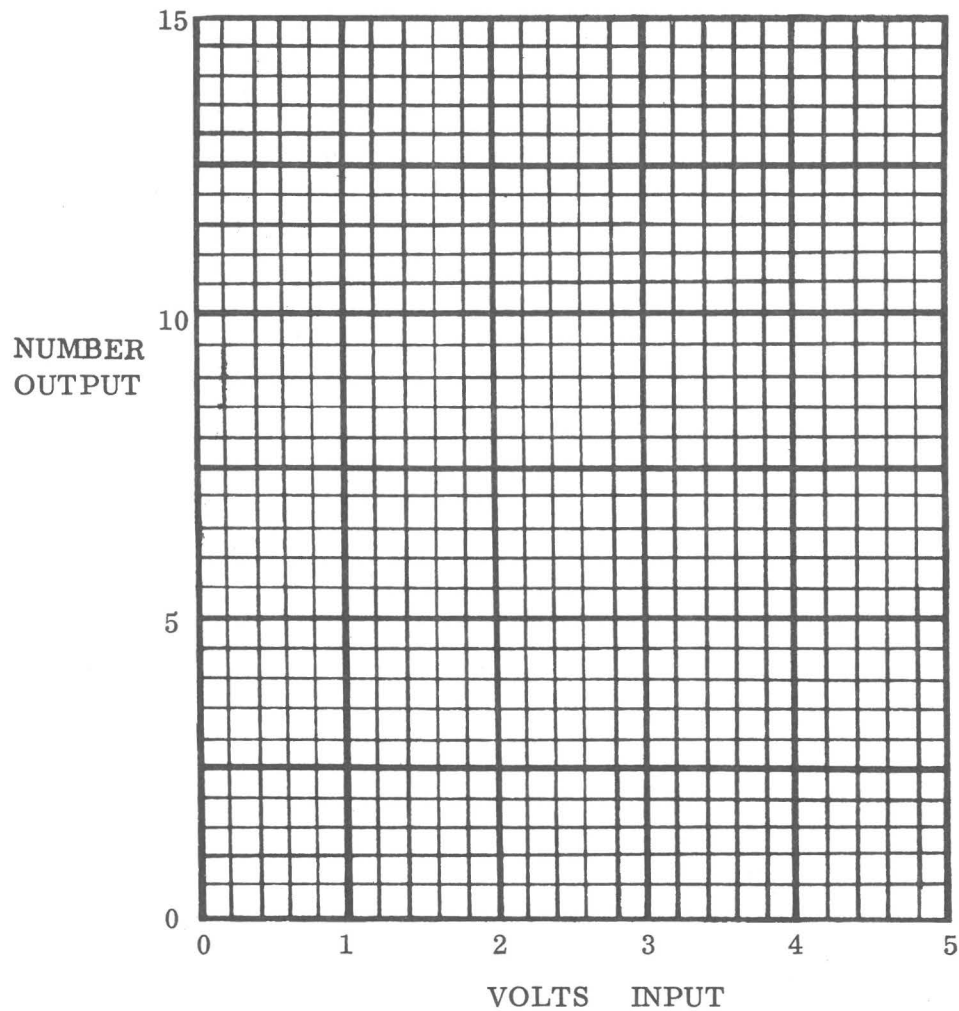
COMPARISON

POT KNOB	VOLTAGE M-DOWN	BINARY NUMBER			
		A	B	C	D
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					

Try using the HALF SPLIT method. Start with 0000, set A up, if the meter reads positive go to B. If the meter reads negative, set A down and go to B. Repeat with B,C, D. The conversion is performed in only 4 steps. Set switch M down to read the voltage X. Set the X potentiometer to all ten dial marks and perform the binary conversion. Enter the data below. Plot the results.

HALF - SPLIT

POT KNOBS	VOLTAGE M-DOWN	BINARY NUMBER			
		A	B	C	D
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					

QUESTIONS:

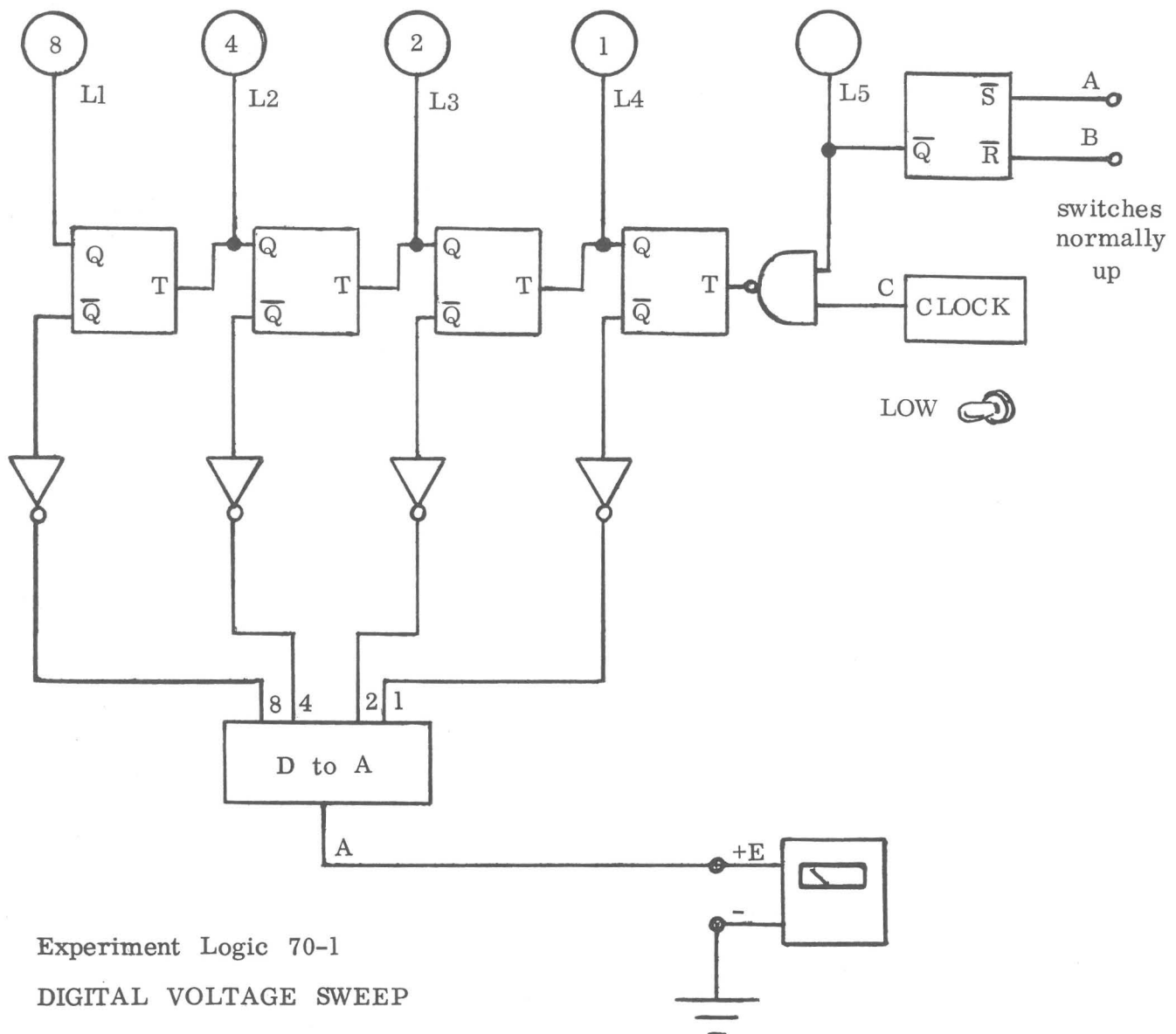
1. Draw the Logic for an A to D Converter using the COMPARATOR instead of the METER. Use the meter to read X voltage.

OBJECTIVE:

A Counter is used to control the D to A converter resulting in a "sweeping" output voltage ramp which is digitally controlled.

DESCRIPTION:

A binary counter is triggered by the clock via a NAND gate. The counter outputs drive INVERTERS which provide accurate voltage sources to the D to A converter. The output voltage sweep is observed on the meter.



Experiment Logic 70-1

DIGITAL VOLTAGE SWEEP

PROCEDURE:

Wire the logic. Observe the output on the meter and record the observations. Stop the sweep at any time using switch A. Use an oscilloscope and the HIGH speed clock position to observe the waveform.

QUESTIONS:

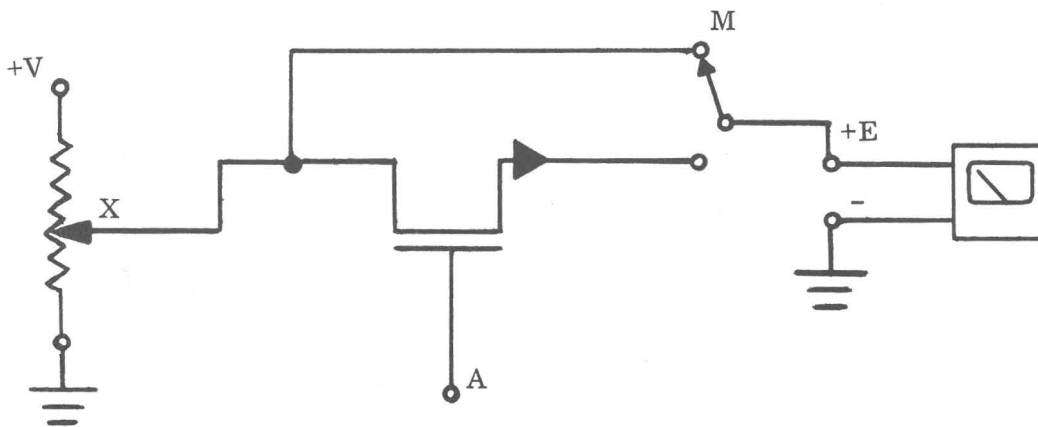
1. What is the lowest output voltage and to what binary count does it correspond?
2. What is the highest output voltage and to what count does it correspond?
3. How long is the "period" or time for a complete sweep?
4. Design the logic for a complete sweep in 8 times (longer than) the above period.
5. What is the frequency of the periodic waveform?

OBJECTIVE:

This experiment illustrates how semiconductor switches are used as "analog voltage gates" to select one of a number of analog voltages.

DESCRIPTION (GATE):

The semiconductor switch is an analog gate.



Experiment Logic 71-1

ANALOG GATE

With control switch A UP, the gate output follows the input (POT X). With Switch A DOWN (0), the output is not affected by X.

PROCEDURE:

Wire the logic and observe the results.

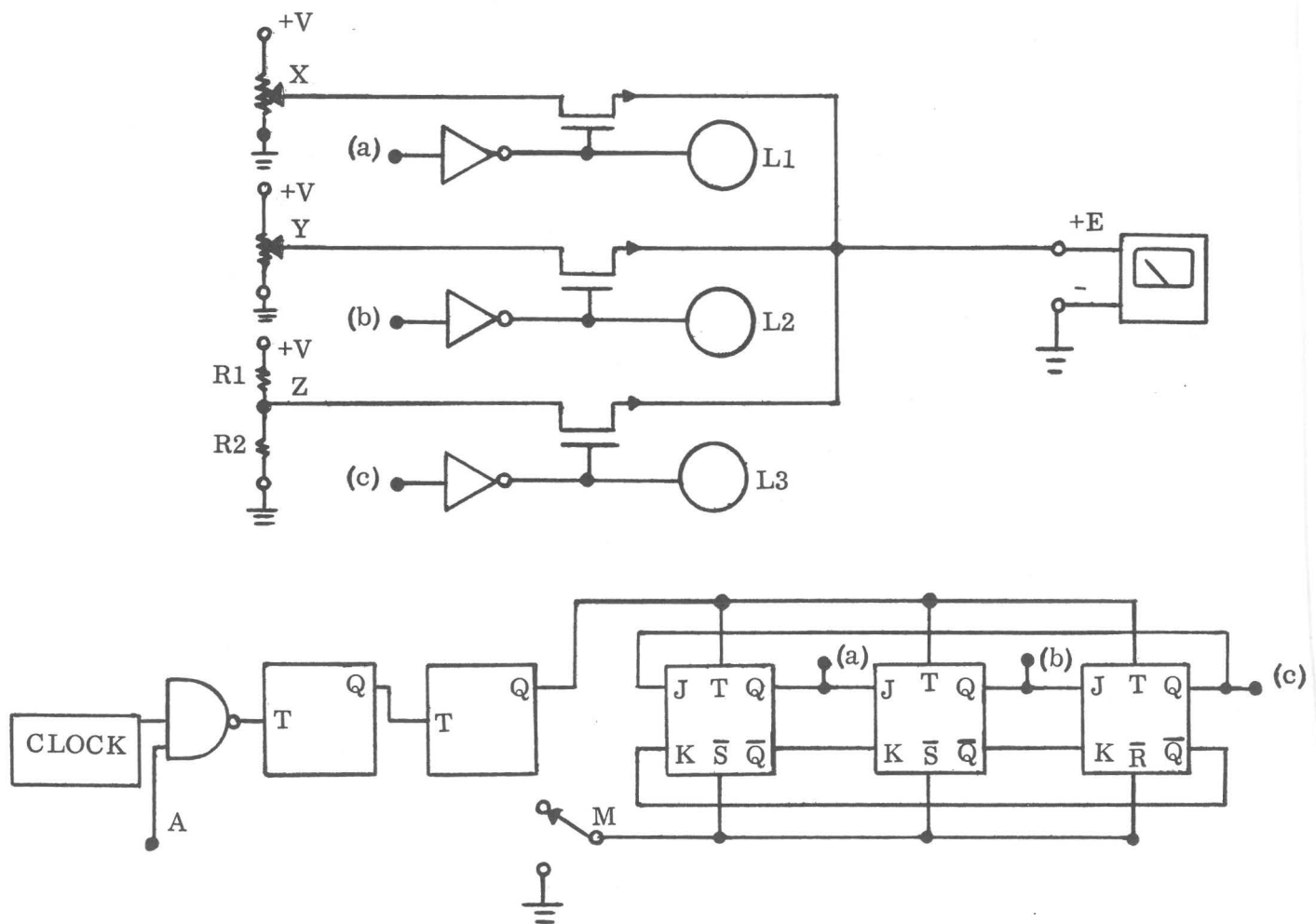
DESCRIPTION (MULTIPLEXER):

Three analog gates are connected to three separate voltage

inputs (X, Y, and the junction of two resistors). The gate control voltage is sequenced from a 3 step RING COUNTER. The gates are sequentially activated and the analog voltages appear sequentially on the meter.

PROCEDURE:

The ring counter is not triggered with switch A DOWN. Depressing switch M sets a 110 into the register. Potentiometers X and Y can be preset for any voltage levels. Set switch A UP and observe the results. Enter the data in the table. Vary the input voltages.



QUESTIONS:

1. Name an application for the multiplex gates.
2. What is the stepping rate for the sampling?
3. What is the sampling rate for each input?

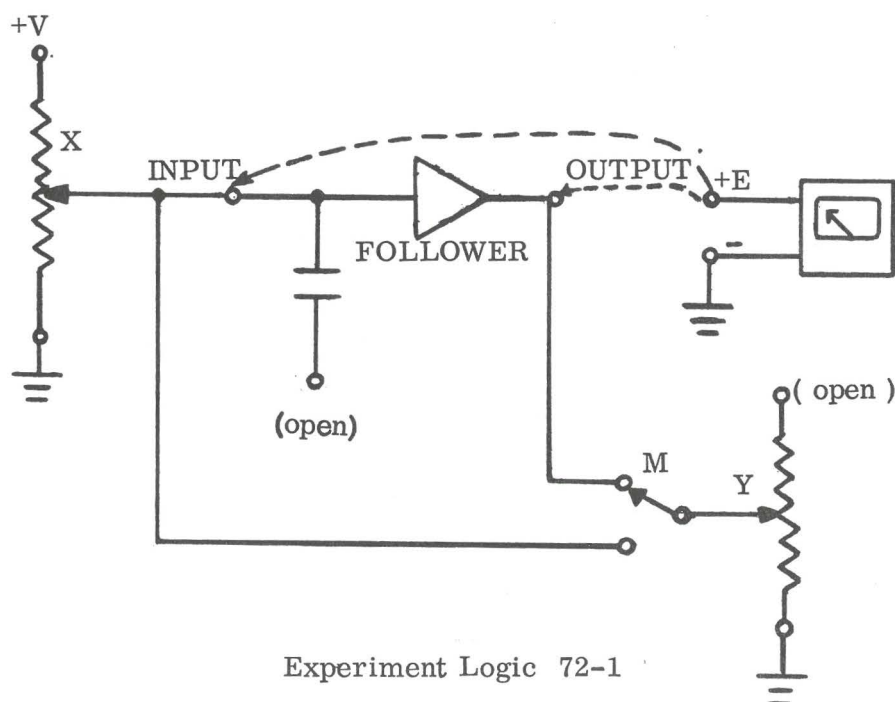
L1	L2	L3	METER VOLTAGE
1	0	0	
0	1	0	
0	0	1	
1	0	0	
0	1	0	
0	0	1	

OBJECTIVE:

This experiment demonstrates the operation of the FOLLOWER or EMITTER FOLLOWER element.

DESCRIPTION (FOLLOWER):

When the center or "wiper" of a potentiometer is "loaded" by a resistance (such as potentiometer Y), the output voltage is decreased. An "emitter follower" (EF) circuit is used as a "buffer" to prevent loading of the circuit. The circuit shown below allows the load (POT Y) to be switched from the input to the output of the emitter follower while the output voltage is monitored.



Experiment Logic 72-1

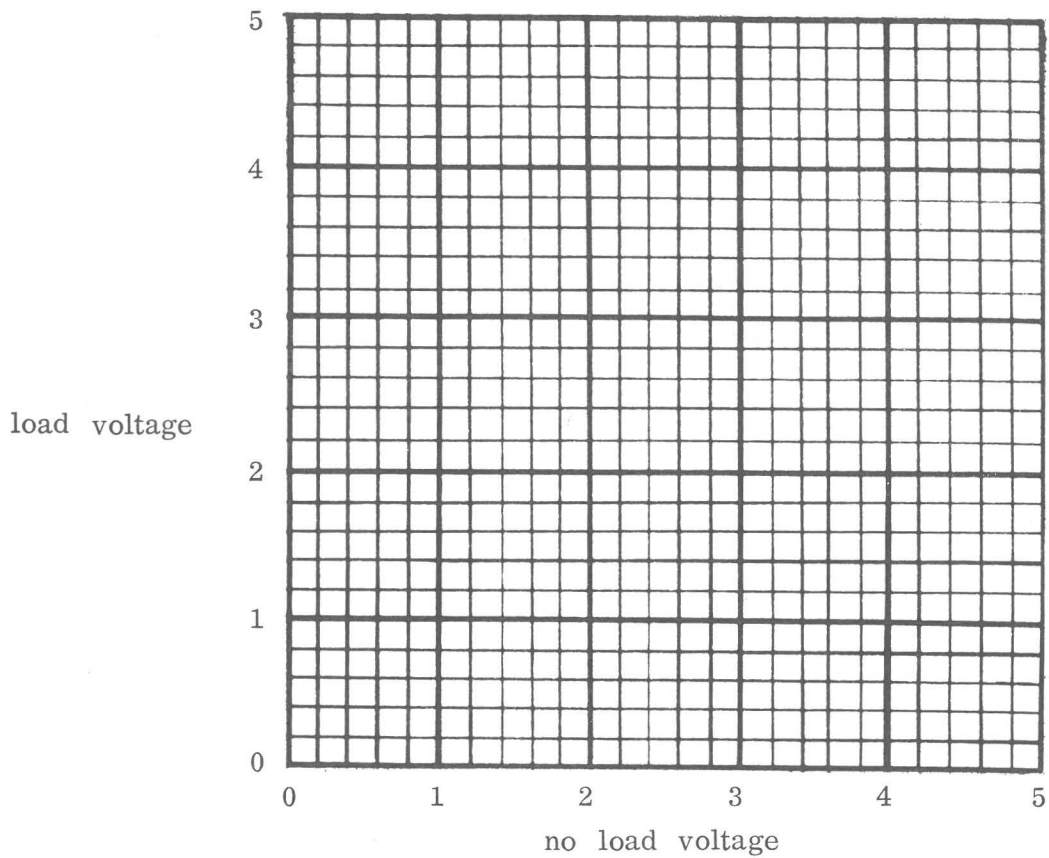
FOLLOWER

PROCEDURE:

Wire the experiment. Set pot X so that the voltage output is about mid scale. Set pot Y so that it is in the center of its range. Connect the meter to the X signal (Follower input). Monitor the input voltage with and without Load Y. Connect the Meter to the FOLLOWER output. Monitor the output voltage with and without the Load. With No-Load, monitor and record

various corresponding input and output voltages.
Enter the Data in the Table below:

POT POSITION	X POT (INPUT)		FOLLOWER OUTPUT	
	VOLTAGE no load	VOLTAGE Y load	VOLTAGE no load	VOLTAGE Y load on follower
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

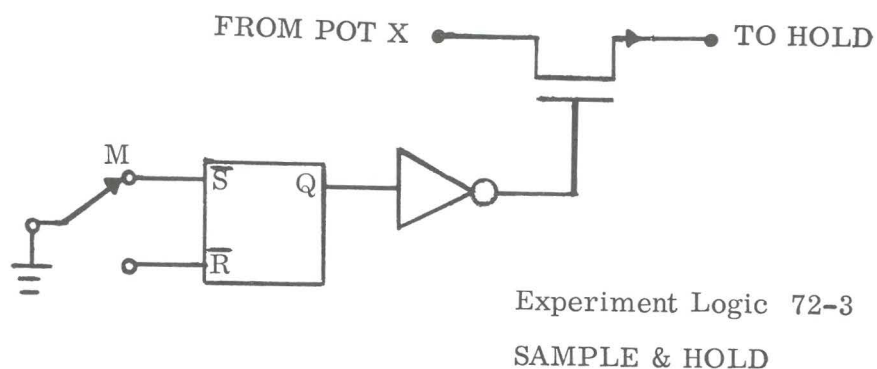
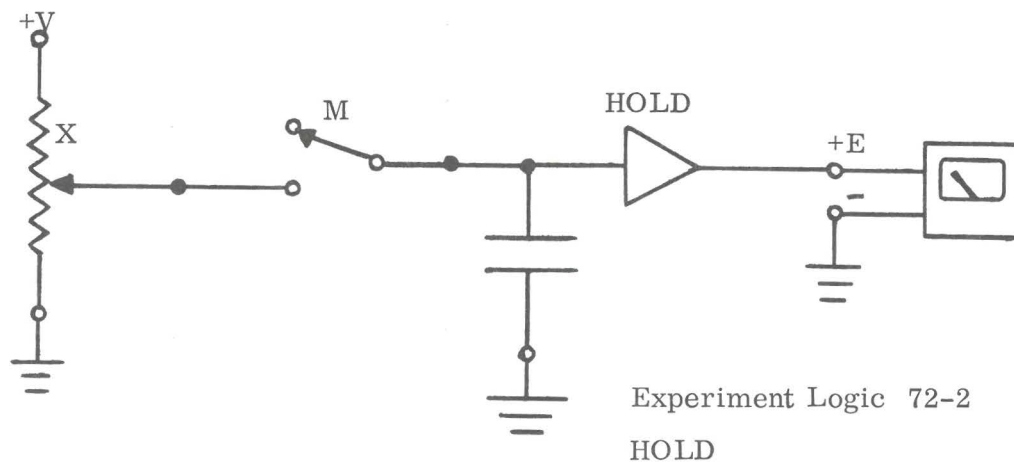


QUESTIONS:

1. What is the follower "gain"?
2. What disadvantage is there in using an emitter follower?
3. What is the "output resistance" of the emitter follower?

DESCRIPTION (HOLD):

When the momentary Switch M is depressed, the analog voltage from potentiometer X is applied to the Follower circuit (Connecting the CAPACITOR to ground results in a HOLD circuit). The capacitor is charged to the input voltage and the follower output corresponds to the capacitor voltage. When the switch is released, the capacitor retains the voltage charge and the output Hold voltage remains at the previous input voltage (with a slow drift rate). Thus the circuit "holds" a voltage when the input is removed. Pot X can be varied and the HOLD output remains constant. Replace switch M with the Analog Gate shown in Figure 72 - 3 to provide a "solid - state" SAMPLE AND HOLD".



PROCEDURE:

Wire the circuit. With switch M up adjust pot X to an intermediate position. Depress switch M and note that the meter registers the input voltage. Release Switch M. The circuit will hold the voltage. Repeat for several input voltages and record the results. Estimate the drift rate. Enter the data in the table below:

VOLTAGE	TIME

QUESTIONS:

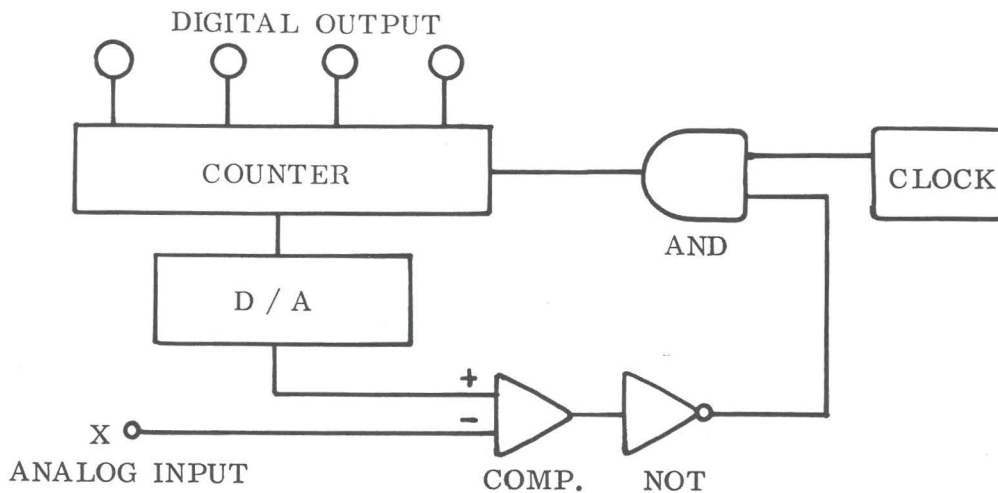
1. Draw the logic for an ANALOG GATE replacement of the Switch M. Use a Flip-Flop to drive a NOT to drive the ANALOG GATE CONTROL input. Connect a lamp to the Analog Gate Control. Trigger the flip-flop with the Clock. This logic is an automatic SAMPLE AND HOLD.

OBJECTIVE:

The previous experiment demonstrated how a D to A Converter was used in a feedback comparison system to implement an Analog to Digital Conversion. This experiment uses the Comparator for an Automatic A to D Conversion. In addition, Calibration is introduced to adjust the output digital numbers to correspond to desired Scale Factors. The Scale Factor is adjusted to read in Volts directly resulting in an instrument used as a Digital Voltmeter.

DESCRIPTION:

The Counter and Comparison technique is employed as shown in the diagram below :



As the count increases, the D/A Voltage Output increases. When the D/A output exceeds the input voltage X , the Comparator output becomes a 1 . This output is converted to a 0 (Not Logic) which is applied to the AND gate to Stop the counter at the proper number.

Later, by inserting another potentiometer Y in either the + or - input paths to the comparator, the voltage point at which the comparator switches is altered. This imposes different scale factors or counts on the Converter System. Adjusting the Scale Factor potentiometer Y with a known Reference Voltage is the Calibration process.

PROCEDURE:

1. Wire the logic as shown in Fig. 73-1. Set various input voltages with potentiometer X and read the voltages on the Meter. Depress Switch M to

clear the counter. When the ON lamp is turned off, the conversion is complete and the digital lamps display the converted digital results. Use a standard binary scale (scale 1) to read the output. Enter various input voltages and digital outputs in DATA TABLE A. Use input voltages at least every 1/2 volt (i.e., 0, 1/2, 1, 1 1/2 volts, etc.).

2. Plot the results (horizontal scale = input voltage ; vertical scale = digital output) . Use Graph A.

3. What is the scale factor of the conversion in counts per volt? _____

4. To adjust the scale factor, modify the logic as shown in Fig. 73-2. The potentiometer Y is driven by the Follower (Hold) element to prevent Loading effects.

5. Set in an input reference voltage just below 3 volts (set 2.8 volts) . Set the Calibration Pot Y at the low end (center tap near ground) . Clear and run the A to D . Slowly keep adjusting Y (center away from ground) until the output reads 3 volts using Scale 2 (i.e., 4, 2, 1, 1/2) resulting in a binary reading of 0110 . The system is now calibrated for scale 2 . Do not alter Pot Y . Record the position of Pot Y . _____

6. Set in various input voltages using input pot X (at least every 1/2 volt) and read the output directly in Volts (not as a binary number) . Use Scale 2 . Enter the results in Table B and Plot the results on Graph B.

7. What is the Resolution of the Voltmeter (minimum readable increment) ? _____.

8. What is the maximum voltage that can be read with the voltmeter?

_____.

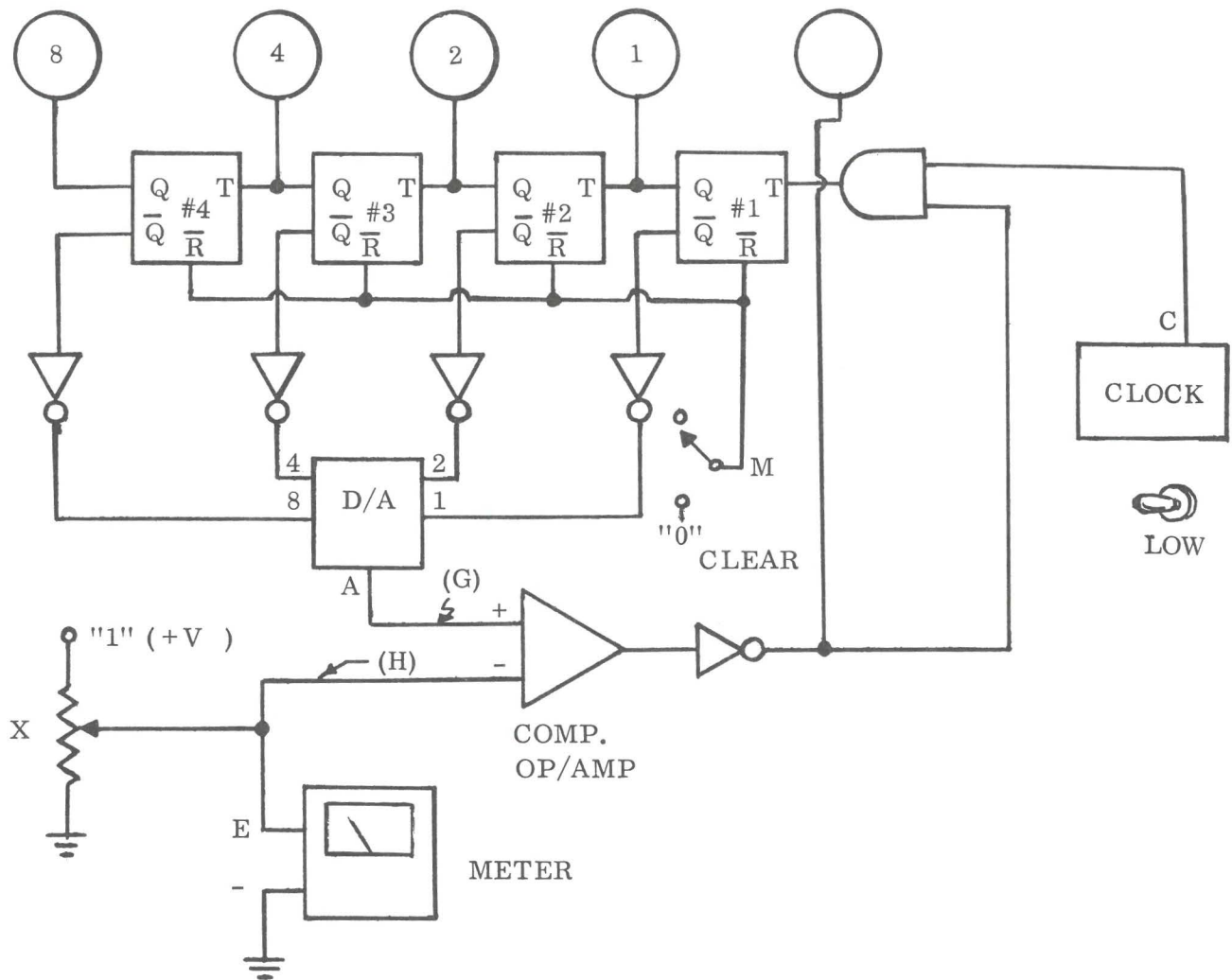
9. What is the dynamic range (ratio of maximum voltage to resolution) ? _____. What is the dynamic range in decibels ? _____.

10. What is the accuracy of the converter ? This is determined by plotting a stepped - line on the graph at the proper scale (not a best fit) and determining the error (select either maximum or average). The Accuracy is expressed as the Error as a percentage of maximum full scale voltage plus or minus 1/2 least significant bit (LSB). The error is sometimes specified as a percentage error plus or minus 1/2 LSB plus any constant Offset .
(All calculations can be deferred until the experiment is completed.)

11. Fig. 73-3 shows the logic required for calibration for a higher (smaller) resolution. Remove the Follower and Pot Y circuit from the X input path and insert it into the D/A path as shown in Figure 3.

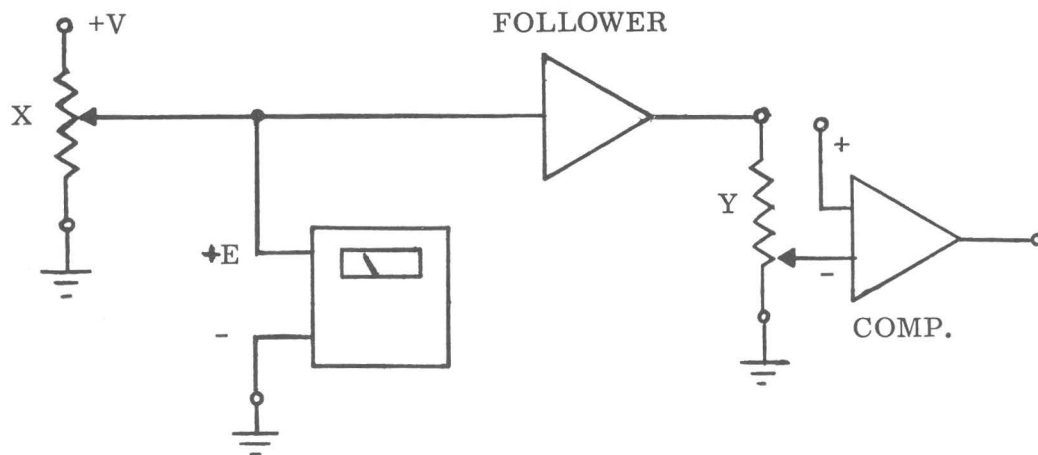
12. Set in a calibration voltage reference below 3 volts (set 2.9 volts) . Set Y at a maximum (center tap furthest from the ground) . Clear and run the converter until it stops. Adjust Y until the output reads 3 volts (i.e. 1100

COUNT	8	4	2	1	ON	SCALE 1
VOLTS	4	2	1	1/2	ON	SCALE 2
VOLTS	2	1	1/2	1/4	ON	SCALE 3



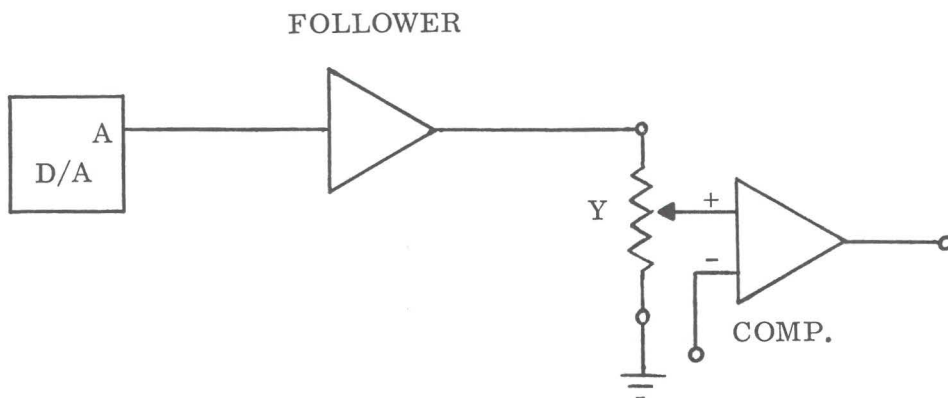
Experiment Logic 73-1

DIGITAL VOLTMETER



Experiment Logic 73-2

CALIBRATION ($1/2$ VOLT)



Experiment Logic 73-3

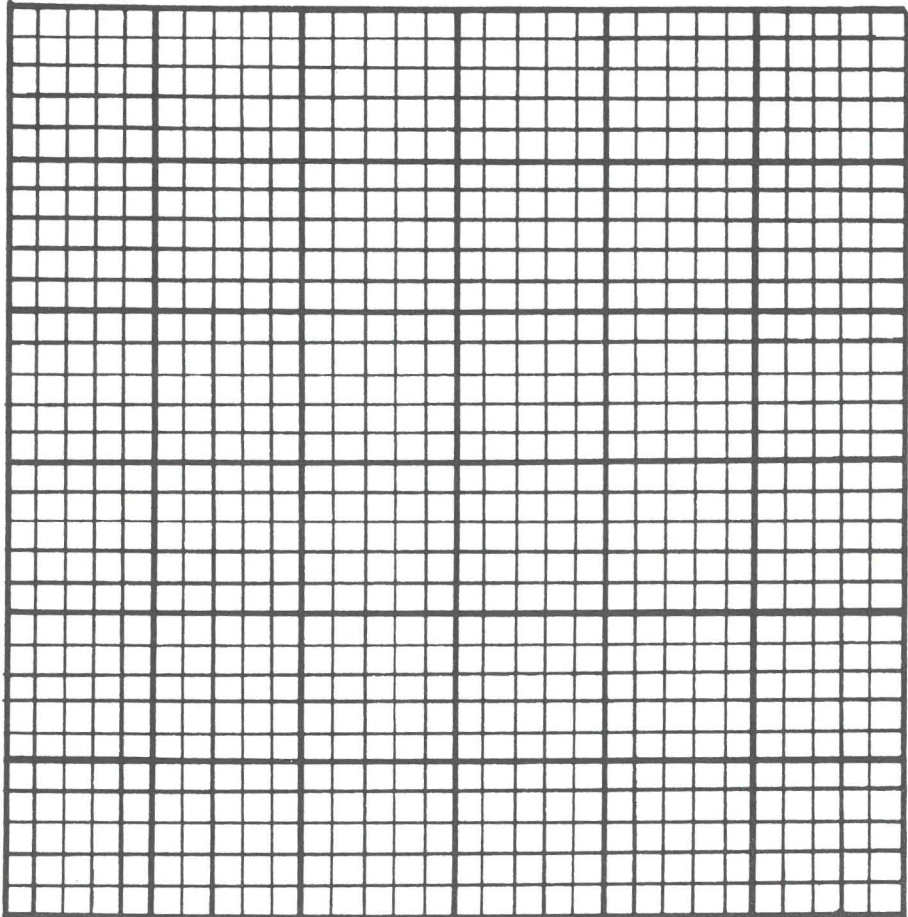
CALIBRATION ($1/4$ VOLT)

DATA TABLE A

INPUT		BINARY NUMBER			
VOLTAGE	8	4	2	1	#

GRAPH

BINARY
NUMBER

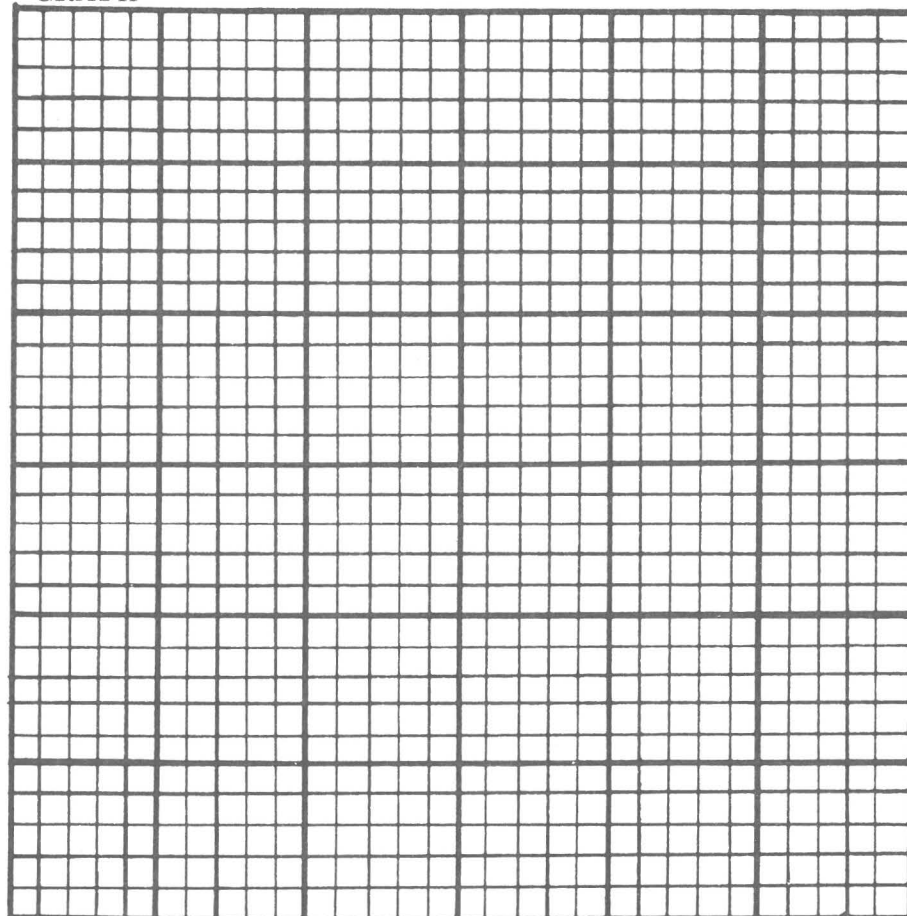


VOLTAGE

DATA TABLE B

INPUT	BINARY NUMBER				
VOLTAGE	8	4	2	1	#

GRAPH



BINARY
NUMBER

VOLTAGE

using Scale 3.) The calibration for scale 3 is now complete and Pot Y should not be altered. Record the position of POT Y. _____

13. Enter various inputs via X and record the Voltmeter outputs directly in Volts using Scale 3. Plot the results. Use Table C and Graph C.

14. What is the maximum voltage range? _____
15. What is the dynamic range? _____
16. What is the resolution? _____
17. What is the Accuracy? _____
18. What is the Conversion Rate in Conversion per Second? _____

QUESTIONS:

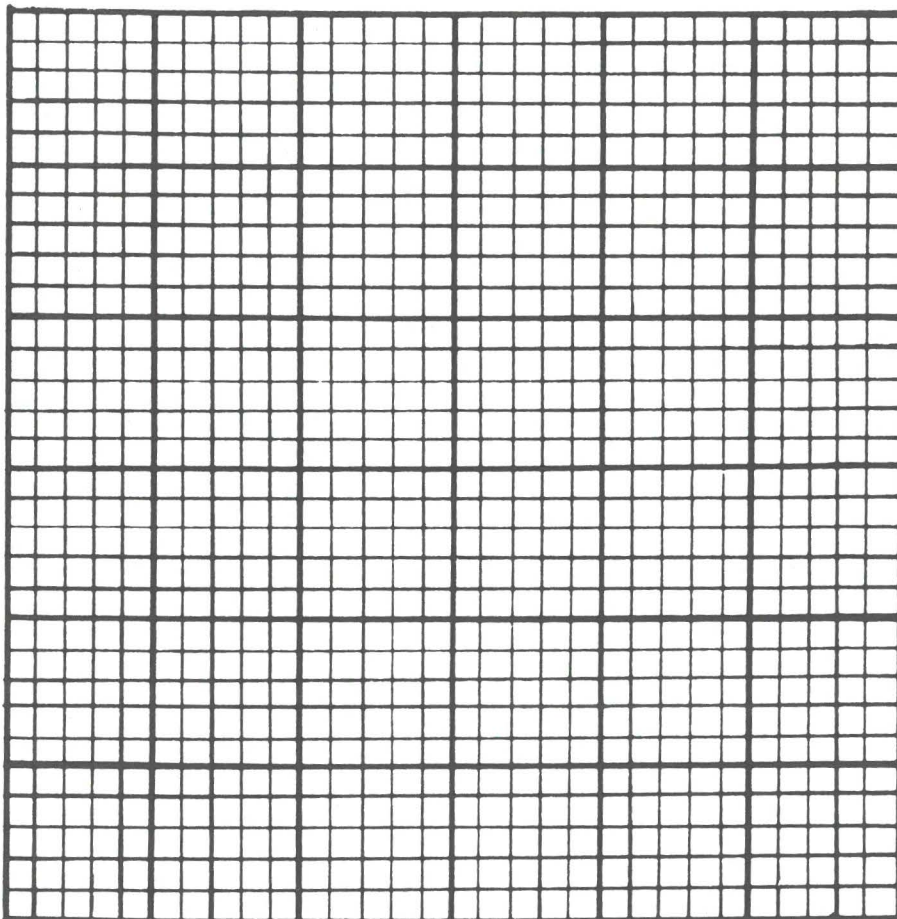
1. What determines the location (input X path or D/A path) of the calibration attenuator for various scale factors?
2. In which path would Y be located for:
 - (a) scale of 8, 4, 2, 1 Volts
 - (b) resolution of 0.1 Volts
 - (c) maximum voltage of 50 Volts
3. What would happen in the Fig. 73-1 A to D Converter for an input of 12 volts?
4. How many binary digits are required for a Digital Voltmeter to read up to 100 volts with a resolution of 0.1 volts?

DATA TABLE C

[illegible]

GRAPH C

BINARY
NUMBER



VOLTAGE

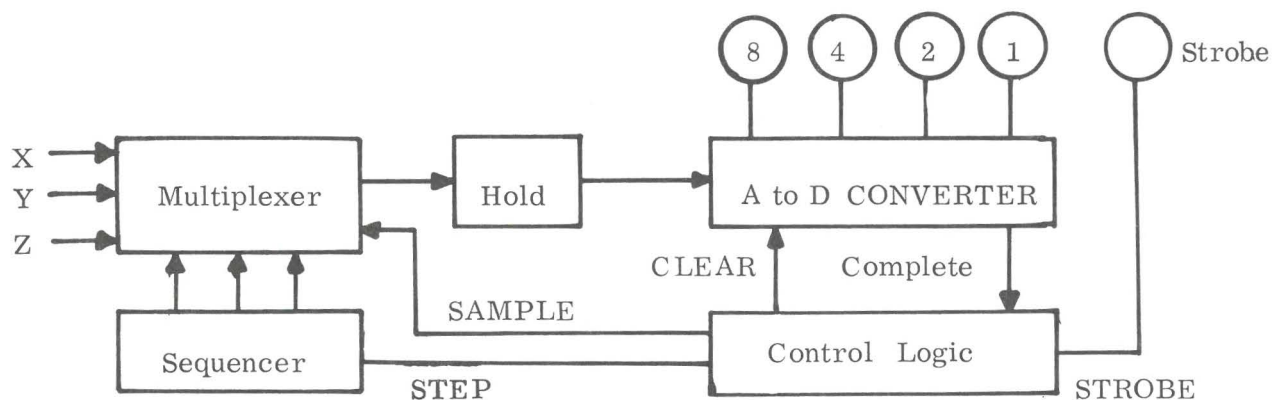
OBJECTIVE:

The previous experiments have demonstrated all of the various conversion, analog gates, control systems using digital logic and the sample and hold element. This experiment combines all of these items into a single system similar to those used in complete data acquisition systems.

DESCRIPTION:

Data Acquisition systems generally have many different analog inputs. It is desired to sample each of these inputs in sequence and convert the voltages to their digital equivalent. The digital number or answer is then transferred to a digital recording or communication system.

The basic elements of this system are shown in the Block Diagram below:



DAS BLOCK DIAGRAM

The Analog Voltage inputs (X, Y, Z) are applied to the Multiplexer which contains three Analog Gates, one for each input. Multiplexers may have 100 inputs. One of the selected analog inputs is supplied to the Sample and Hold. A SAMPLE signal performs the function of opening the selected gate for the short sample time. In some systems, a separate sample gate is used. The proper analog gate in the multiplexer is selected in sequence by a Sequencer which contains memory and keeps track of the last input voltage channel selected. The Hold circuit holds the voltage at the input to the Analog to Digital Converter. A COMPLETE signal from the A to D Converter signals a Control Logic System. This logic then generates a STROBE signal to the digital recording system to store the converted data (digital 8421 outputs). The control logic then generates a STEP signal to step the sequencer to the next channel select. A CLEAR signal is then generated to reset the A to D Converter for the next input and the SAMPLE signal is again generated sample the next input. It is of course important that everything be done in the proper order. This is the function of the control logic.

LOGIC DIAGRAM:

The Logic Diagram for the system is shown in Fig. 74-1. The use of tie points is indicated for convenience. The familiar A to D Converter is shown using the counter, D to A Converter, comparator and AND gate control of the clock into the counter. However, in this case, instead of the Comparator feeding back to the AND gate, an intermediate flip-flop (# 9) is used . This is done to provide for later automatic clearing of the A to D . The output of Flip-Flop # 9 signals the STROBE for the external digital system indicating that conversion is complete. The external system will then sample the data and the sequencer address specifying the channel corresponding to the data. Flip-Flop # 9 also generates a HALT signal to shut off the AND gate with the clock pulses to the A to D converter counter.

The Sequencer is a two stage counter (Flip-Flops # 7 and # 8) implemented using the Shift Register Counter (or Johnson Counter) technique (shift register with reversed end around connection). The HALT line becomes a logical zero to halt the AND gate and also triggers the sequence counter to the next step.

The Analog inputs are provided by means of potentiometers X , and Y , and the two Resistors. Thus X and Y are variable, and Z is fixed at some mid value of voltage. Z could be variable by the operator by connection of different resistors in the voltage divider combination in the threshold resistor circuit. The three solid-state analog gates are the multiplexer gates . AND gate inputs to these gates decode the proper count of the sequencer. The sequencer counts are:

00	Standby
10	X Channel
11	Y Channel
01	Z Channel

Thus after the Z Channel Sample, the next A to D Conversion is performed on the same channel. This is the fixed channel and serves as a reference to the operator and provides time to vary X and Y .

The SAMPLE signal is generated by the One-Shot. Two resistors are used to provide a wider sampling pulse. The output of the one-shot is normally a zero. When the HALT signal goes to zero, the Sample pulse is generated. The Sample pulse is connected to the AND gates which operate the multiplexer gates.

All multiplexer outputs are connected to the Hold circuit and the voltage sampled is stored in the Hold Capacitor . The output of the Hold circuit is connected to the Comparator negative (-) input. The meter also monitors the Hold output voltage.

Flip-Flop # 9 is connected to Flip-Flop # 10 in order to provide a CLEAR and restart for the A to D Converter. Flip-Flop # 10 is connected as a shift register to FF # 9. Thus in the next Clock (trigger) time (approx 1 second) after the STROBE signal, FF # 10 output Q becomes a 1 . This output is the CLEAR signal. It clears the A to D Counter as well as FF # 9 . At the next clock pulse time, FF # 10 becomes the same as FF # 9 which has just been cleared, and thus FF # 10 Q output becomes one and the CLEAR signal is gone. The Clear becomes a zero to be active. The entire process is then repeated.

PROCEDURE:

Wire the Logic as shown in Fig. 74-1. Note that the Threshold Logic resistor network is used to provide the voltage divider resistors . Switch M is normally up and provides a manual CLEAR for the system . Note that A and B are the sequencer outputs.

The system runs by itself. Strobe lamp # 5 indicates when the data is valid.

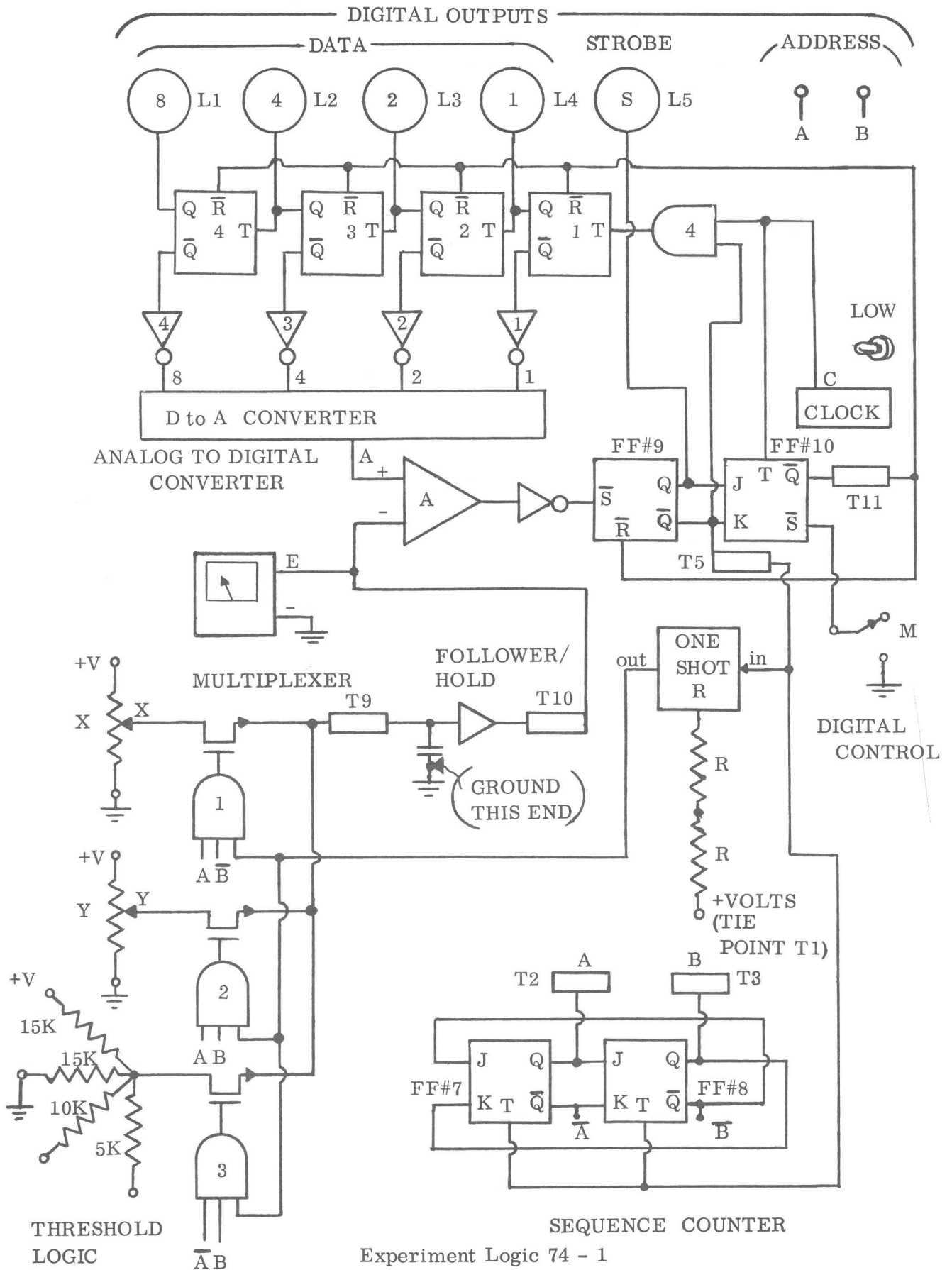
3. Adjust X and Y and select combinations of resistors for Z .
Use the table for Channels 1, 2, 3 , (recall that channel 3 appears twice in a row for reference). Record the analog voltage for each channel. On a second sequence, look for the STROBE lamp and record the digital outputs for each channel.

Channel	Voltage	Digital Output
1		
2		
3		
1		
2		
etc.		

QUESTIONS:

1. Draw a complete logic diagram for a Data Acquisition System with 10 (ten) analog inputs and a 10 binary digit Analog to Digital Conversion. Do not draw the block diagram instead. The inputs are thus, 10 analog input voltages and an external input clear signal. The outputs are the 10 binary bits, the Sequence counter address, and the Strobe pulse. You must determine how many bits to use in the sequence counter and the type of counter and decoding to use.

2. Tabulate the number of elements (Flip-Flops, AND gates, etc.) that are used. At the average power dissipation of 5 milliwatts per element, calculate the total power. Compare this power to an ordinary 100 watt bulb .



Experiment Logic 74 - 1
DATA ACQUISITION SYSTEM

OBJECTIVE:

This experiment uses the electronic components in the Ed-Lab Trainer to demonstrate the circuits used in various types of Integrated Circuit construction. These circuits include: DTL (diode transistor logic), TTL (transistor transistor logic), DL (diode logic), RTL (resistor transistor logic).

DESCRIPTION:

DIODE LOGIC is the most common form of logic used with discrete components. Diodes are used for the AND and the OR Logic functions. Transistors are not used and this type of logic is the least expensive for standard component circuits. Transistors are used for the NOT circuit. The OR gate, however, can not be used to "drive" any other logic element but must be followed by a NOT. Diode logic is shown in Fig. 75-1, 2, 3.

DTL or DIODE - TRANSISTOR LOGIC consists of a Diode AND followed by a Transistor NOT . The Logic element is therefore a NAND . The DTL element is very popular in Integrated circuits. DTL Logic is shown in Figure 75-4.

TTL or TRANSISTOR - TRANSISTOR LOGIC is the most popular form of integrated circuit logic. It consists of an input transistor with many emitters for the inputs as shown in Fig. 75-5. The basic logic element is that of the NAND. When other logic functions are desired, the combinations of NANDs are used.

RTL or RESISTOR-TRANSISTOR LOGIC consists of configurations of two types, both based on the NOT circuit shown in Fig. 75-3. A NOT circuit with many resistor inputs is an RTL NOR element used with circuit components. The advantage over the DTL is that diodes are not required and there is a cost savings. This circuit is shown in Fig. 75-6. The NOT circuit is also used in parallel with many outputs connected together to form the NOR as shown in Fig. 75 - 7.

ECL or EMITTER - COUPLED LOGIC consists of a bias transistor and input transistors coupled via their emitters. The transistors are never switched on to "saturation" and thus the ECL element provides very high speeds of operation. The ECL Logic performs the NOR function. This logic circuit is shown in Figure 75 - 8.

HTL or HIGH THRESHOLD LOGIC is used primarily in industrial applications where there is a lot of noise. The level for a logic 1 required to activate the circuit is double or triple that in the ordinary circuits to exclude the effects of noise on the lines. This is normally accomplished by diode voltage drops in series with the signal inputs. Typical HTL Logic is shown in Fig. 75 - 9. The cost of this logic is usually higher than the others.

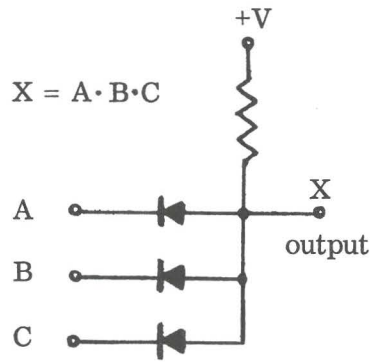


Fig. 75-1 DL AND

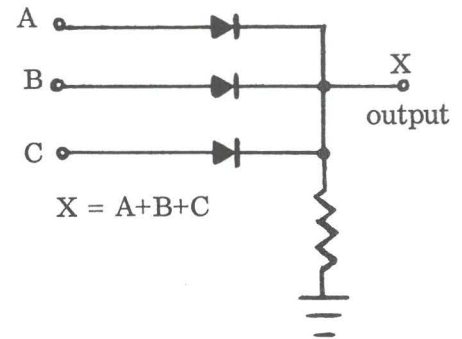


Fig 75-2 DL OR

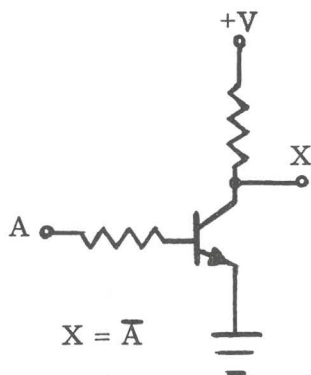


Fig. 75-3 NOT

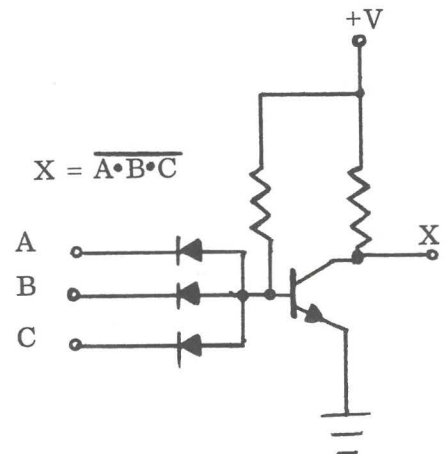


Fig. 75-4 DTL NAND

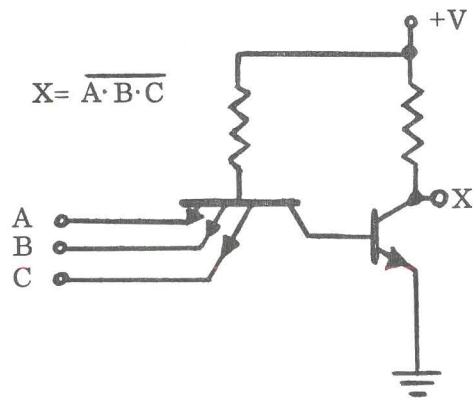


Fig 75-5 TTL NAND

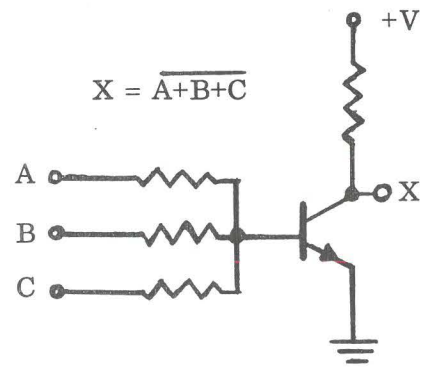


Fig. 75-6 RTL NOR

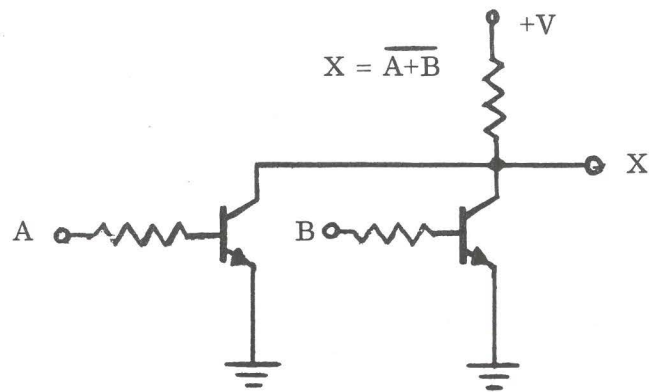


Fig. 75- 7 RTL NOR

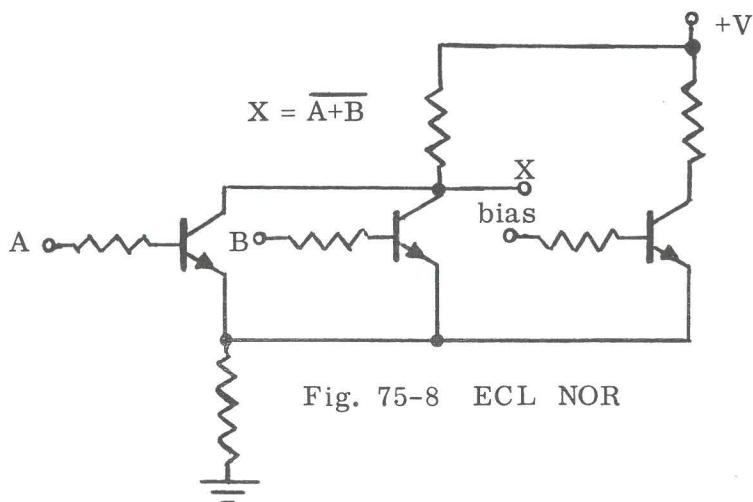


Fig. 75-8 ECL NOR

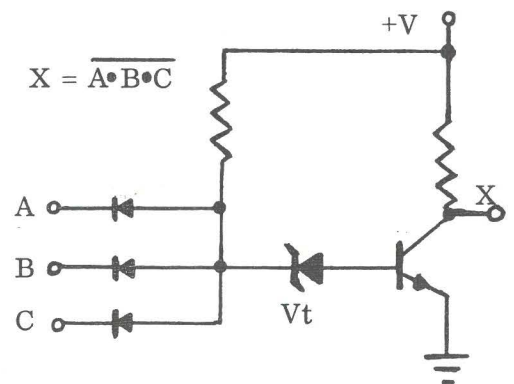


Fig. 75-9 HTL NAND

PROCEDURE:

1. This experiment consists of wiring the circuits for each of the nine Integrated Circuits shown in Fig. 75-1 thru 9. The input switches are then set for the various combinations of inputs and a Truth Table is prepared for each of the circuits. The available circuit components are shown in Fig. 75-10.

2. Wire the Diode Logic AND circuit in Fig. 75-11 with 2 or 3 inputs. Prepare the Truth Table. Is the output an AND ? _____

3. Wire the Diode Logic OR in Fig. 75-12 Prepare the truth table.

4. Wire the NOT circuit in Fig. 75-13. Prepare the Truth Table.

5. Connect the OR gate to the NOT and prepare a truth table.

6. Wire the DTL NAND and prepare the Truth Table. (Fig. 75-14).

7. Wire the TTL NAND in Fig. 75-15 and prepare the Truth Table.

8. Wire the RTL NOR in Fig. 75-16 and Prepare a Truth Table.

9. Wire the RTL NOR in Fig. 75-17 and prepare a Truth Table.

10. Wire the ECL circuit in Fig. 75-18 and Prepare the Truth Table.

11. Wire the HTL NAND in Fig. 75-19. This circuit contains a semiconductor junction drop and a voltage divider to provide a voltage threshold. The input is supplied via a potentiometer. One of the standard Logic NOT Logic elements is also used (DTL). Vary the input voltage and determine the voltage threshold at which the output switches for both circuits. Compare the standard DTL element and the HTL Logic.

EXERCISES:

1. Review the actual data sheets for the various integrated circuits. Note that in many cases there are a number of output transistors to provide output power. The actual logic used in the trainer is the DTL MC 930/830 series shown in some of the data sheets.

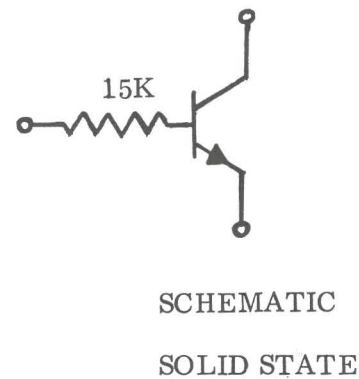
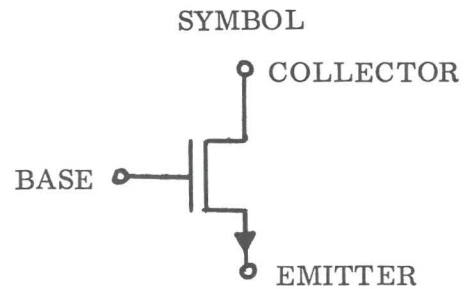
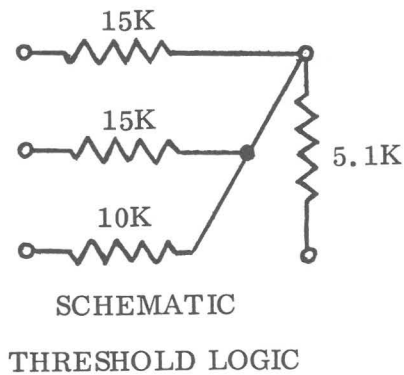
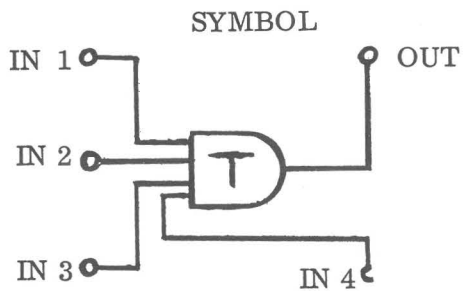
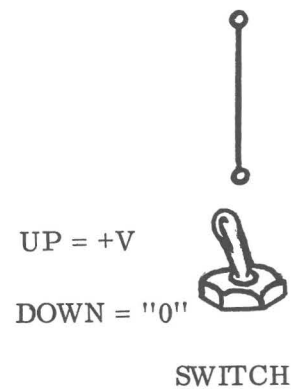
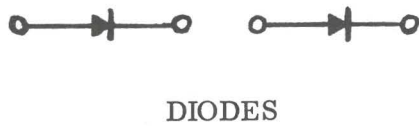
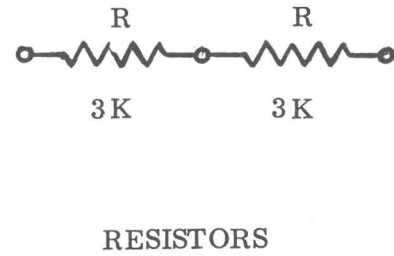
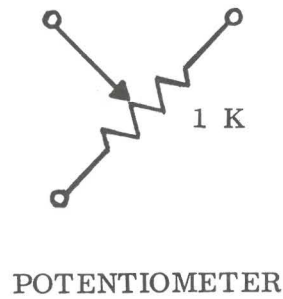
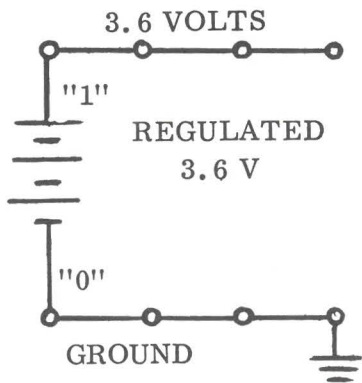


FIG. 75-10 COMPONENTS

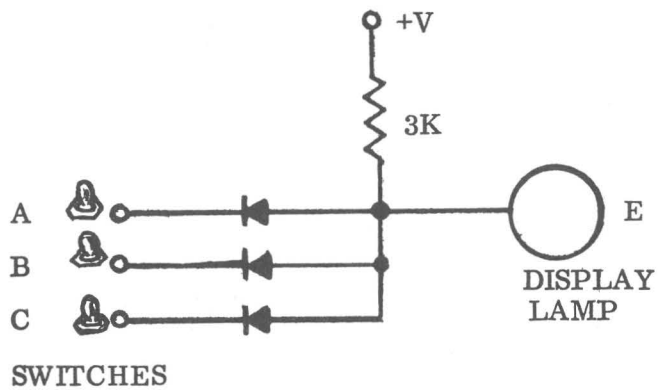


Fig. 75 - 11 DL AND

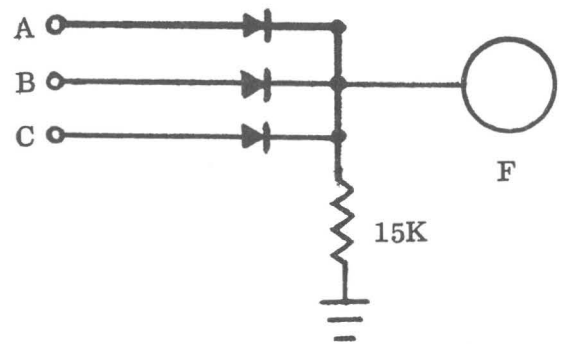


Fig. 75 - 12 DL OR

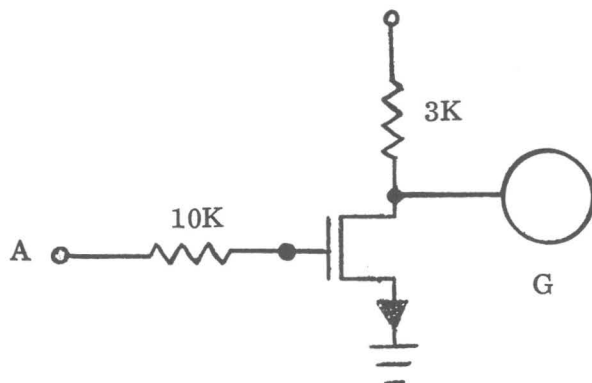


Fig. 75 - 13 NOT

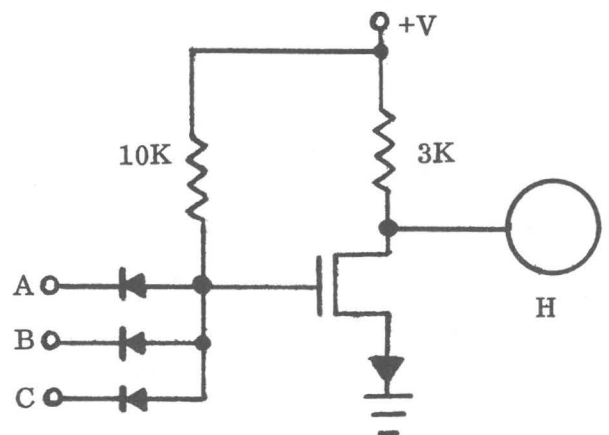


Fig. 75 - 14 DTL NAND

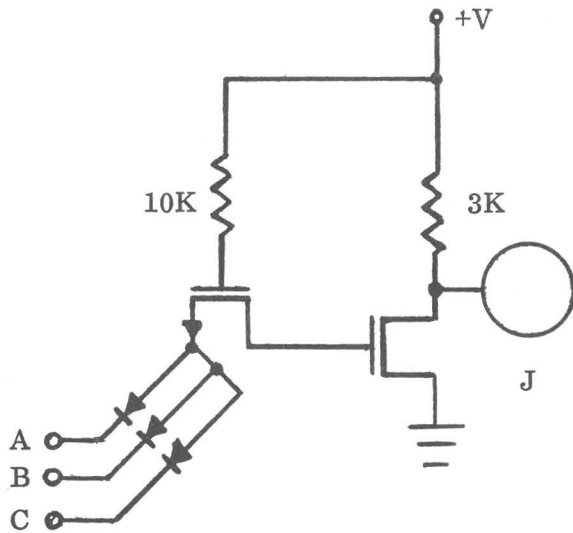


Fig. 75 - 15 TTL NAND

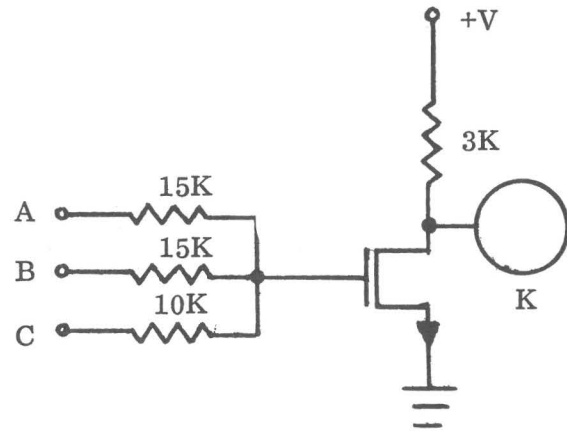


Fig. 75 - 16 RTL NOR

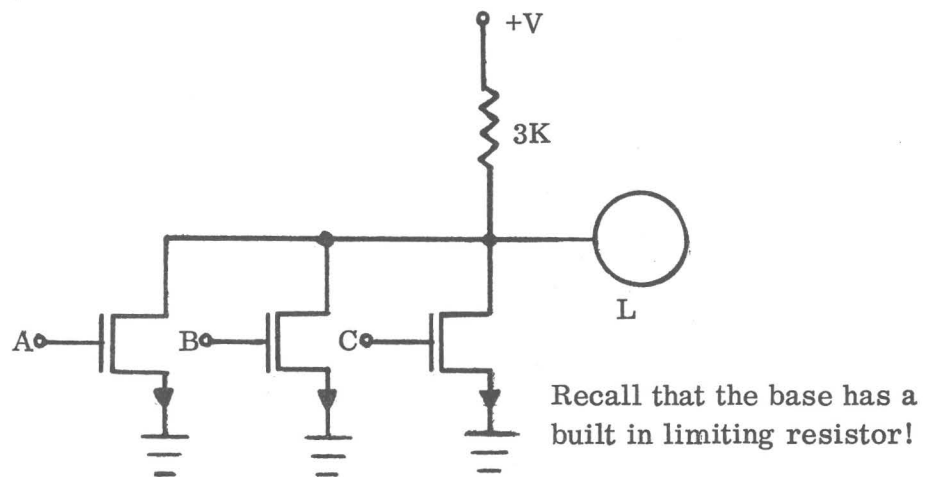


Fig. 75 - 17 RTL NOR

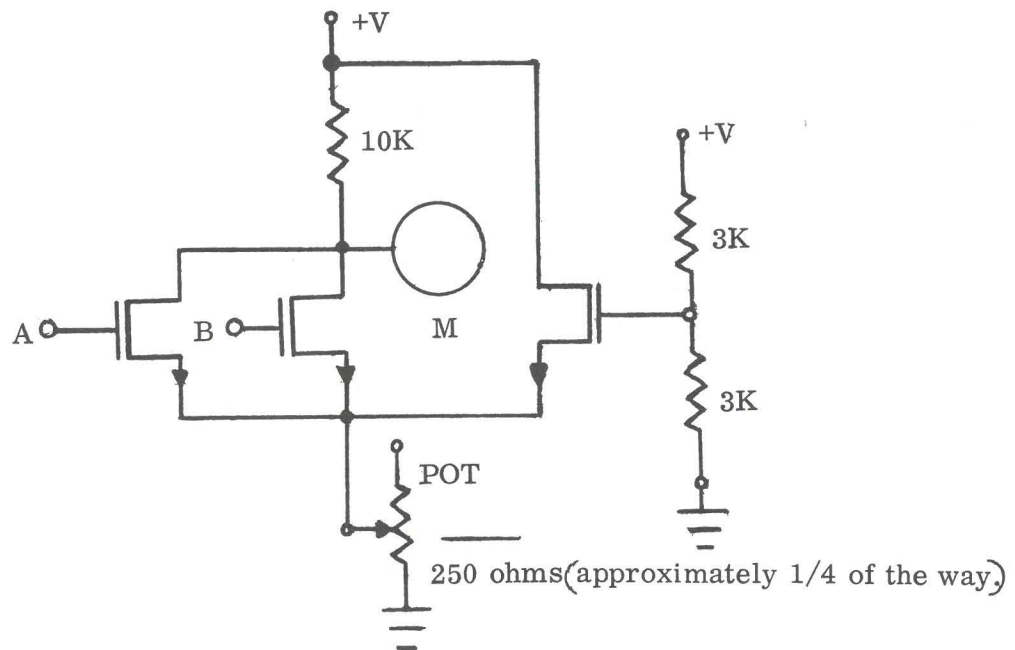


Fig. 75 - 18 ECL NOR

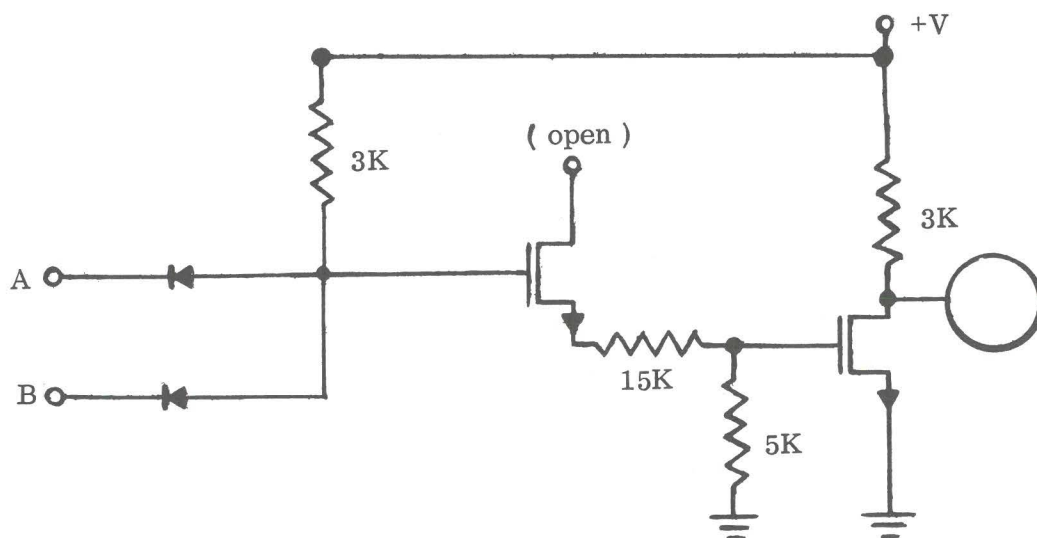


Fig. 75 - 19 HTL NOT

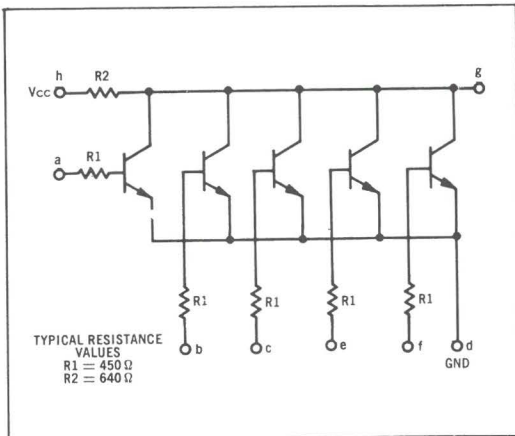
5-INPUT GATES

MRTL MC900/800 series

MC929 • MC829

Available in TO-99 Metal Can, Add "G" Suffix.
Available in TO-91 Flat Package, Add "F" Suffix.

Provides positive logic NOR function. Individual gates may be paralleled with other logic elements for increasing the number of inputs (subject to loading rules).



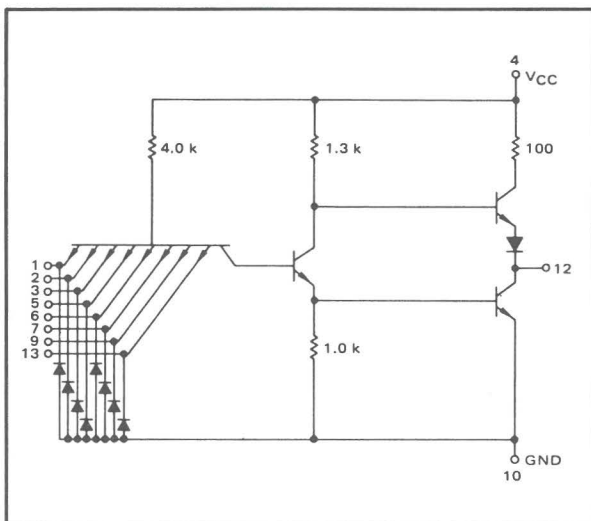
PIN CONNECTIONS

SCHEMATIC	a	b	c	d	e	f	g	h
G PACKAGE (TO-99)	1	2	3	4	5	6	7	8
F PACKAGE (TO-91)	2	3	4	5	7	8	9	10

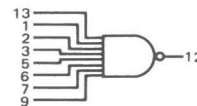
SINGLE 8-INPUT "NAND" GATE

MTTL MC500/400 series

**MC502 • MC552
MC402 • MC452**



This device is an 8-input NAND gate. It is useful when processing a large number of variables, such as in encoders or decoders.



Positive Logic:
 $12 = 1 \cdot 2 \cdot 3 \cdot 5 \cdot 6 \cdot 7 \cdot 9 \cdot 13$
Negative Logic:
 $12 = 1 + 2 + 3 + 5 + 6 + 7 + 9 + 13$

Total Power Dissipation = 15 mW typ/pkg
Propagation Delay Time = 12 ns typ

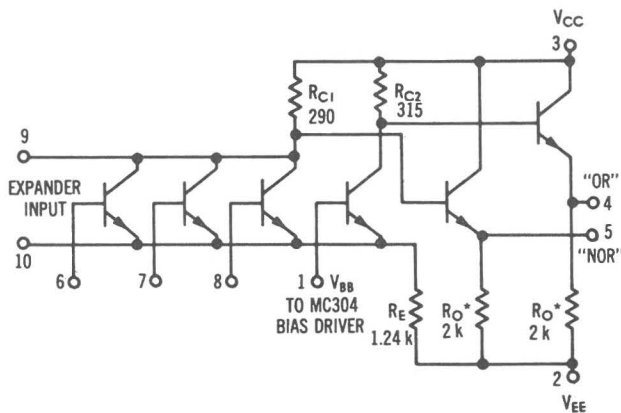
SERIES	INPUT LOADING FACTOR (I_F)	OUTPUT DRIVE (I_{OL})	TEMPERATURE RANGE
MC502 MC552	1 (-1.33 mA)	15 MC500 series Gates (20 mA) 7 MC500 series Gates (10 mA)	-55°C to +125°C
MC402 MC452	1 (-1.66 mA)	12 MC400 series Gates (20 mA) 6 MC400 series Gates (10 mA)	0° to +75°C

3-INPUT GATES

MECL MC300 series

MC306 • MC307

Expandable 3-input gates that provide the positive logic "NOR" function and its complement simultaneously. MC307 omits output pull-down resistors, permitting reduction of power dissipation.

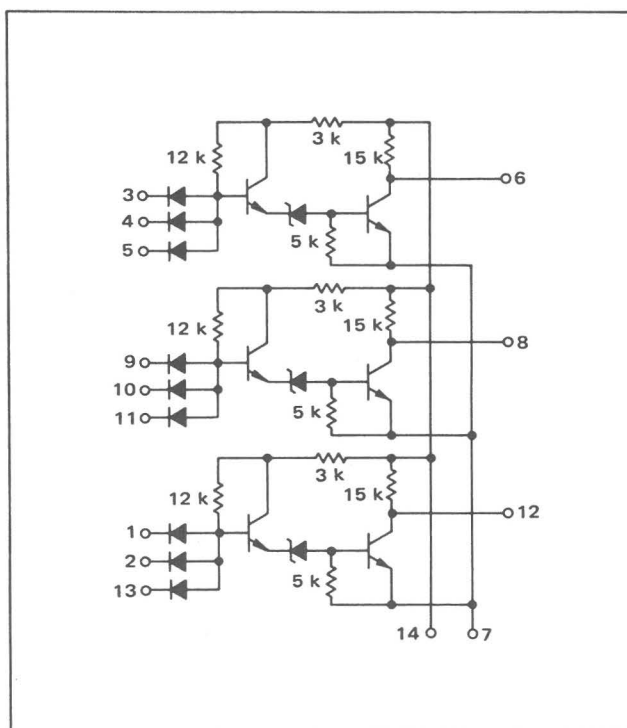


*Resistors R_O are omitted in MC307 circuits to permit reduction of Power Dissipation in systems where logic operations are performed at circuit outputs.

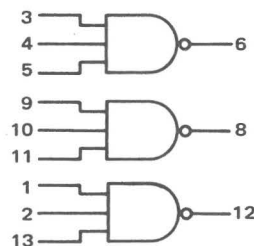
TRIPLE 3-INPUT GATE

MHTL MC660 series

MC670P



This device consists of three 3-input NAND gates with passive output pull-up.



Positive Logic: $6 = \overline{3 \cdot 4 \cdot 5}$

Input Loading Factor = 1

Output Loading Factor = 10

Propagation Delay Time = 125 ns typ

Typical Total Power Dissipation

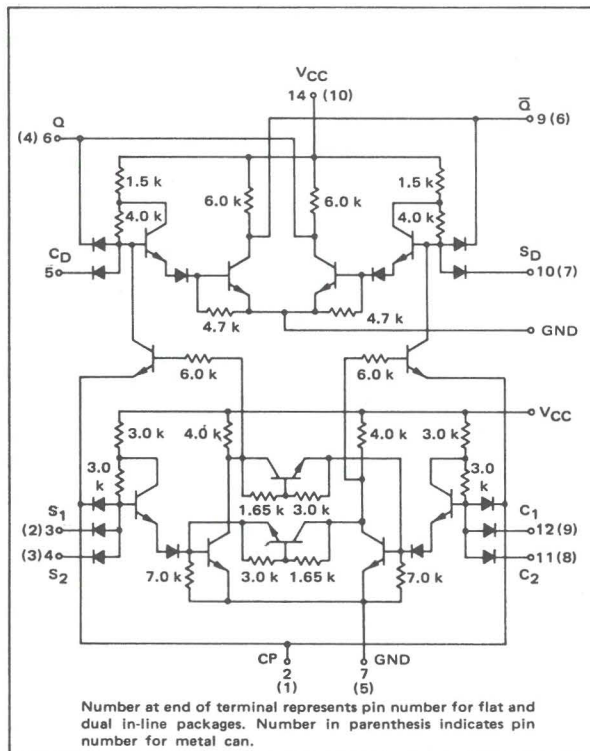
Input High = 132 mW

Inputs Low = 39 mW

CLOCKED FLIP-FLOPS

MDTL MC930/830 series

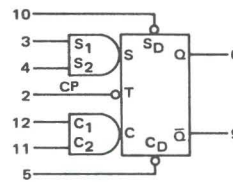
MC931F, G • MC831F, P, G



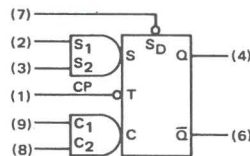
This clocked flip-flop consists of two directly coupled flip-flops, operating on the "master-slave" principle. Operation depends only on voltage levels, and the shape of the input clock becomes unimportant in determining the state of this flip-flop. The input information is stored in the "master" flip-flop when the clock voltage is high, and is transferred to the "slave" when the clock voltage is low.

This clocked flip-flop can be operated in either the R-S or J-K mode. For J-K operation the Q output is connected to a clear input, and the $\bar{Q}$ output is connected to a set input. Direct set (S_D) and direct clear (C_D) inputs are available.

MC931F/MC831F, P



MC931G/MC831G



SYNCHRONOUS TRUTH TABLE

S_1	t_n			t_{n+1}
	S_2	C_1	C_2	Q
0	X	0	X	Q_n
0	X	X	0	Q_n
X	0	0	X	Q_n
X	0	X	0	Q_n
0	X	1	1	0
X	0	1	1	0
1	1	0	X	1
1	1	X	0	1
1	1	1	1	U

0 - Low State (more negative)

1 - High State (more positive)

X - State of the Input does not affect the state of the circuit.

U - Indeterminate State

J-K TRUTH TABLE
(Connect S_2 to $\bar{Q}$, C_2 to Q)

S_1	t_n		t_{n+1}
	C_1	Q	
0	0	Q_n	Q_n
1	0	1	1
0	1	0	0
1	1	$\bar{Q}_n$	$\bar{Q}_n$

ASYNCHRONOUS TRUTH TABLE

S_D	C_D	Q	$\bar{Q}$
1	1	NC	NC
0	1	1	0
1	0	0	1
0	0	1	1

Asynchronous inputs, direct set (S_D) and direct clear (C_D), override the synchronous inputs; they are independent of all other inputs.

Input Loading Factor:

S and $C = 2/3$

S_D and $C_D = 3/4$

$T = 2$

Output Loading Factor = 7

Total Power Dissipation = 55 mW typ/pkg

Propagation Delay Time = 40 ns typ

MDTL MC930/830 series

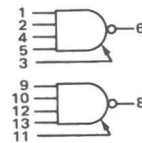
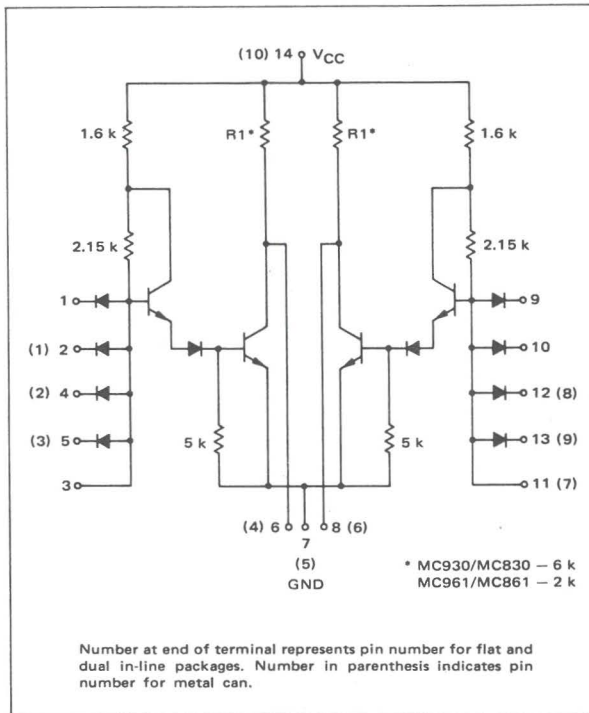
EXPANDABLE DUAL 4-INPUT GATES

MC930F • MC830F, P
MC961F • MC861F, P

EXPANDABLE DUAL 3-2 INPUT GATES

MC930G • MC830G
MC961G • MC861G

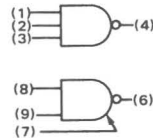
This gate element, in the 14-pin flat and dual in-line packages, consists of two expandable 4-input NAND gate circuits. Since the metal can (G suffix) has only 10 pins, that circuit consists of one 3-input and one 2-input expandable gate. The elements may be cross-coupled to form a bistable multivibrator, or the outputs may be connected in parallel to perform the logic "OR" function.



MC930F/MC830F, P
MC961F/MC861F, P

Positive Logic: $6 = \overline{1 \cdot 2 \cdot 4 \cdot 5 \cdot [3]}$

Negative Logic: $6 = 1 + 2 + 4 + 5 + [3]$



MC930G/MC830G
MC961G/MC861G

Positive Logic: $4 = \overline{1 \cdot 2 \cdot 3}$

Negative Logic: $4 = \overline{1 + 2 + 3}$

Input Loading Factor = 1

Output Loading Factor:

$$\text{MC930/MC830} = 8$$
$$\text{MC961/MC861} = 7$$

Total Power Dissipation:

MC930/MC830 = 22 mW typ/pkg

MC961/MC861 = 33 mW typ/pkg

Propagation Delay Time

MC930/MC830 = 30 ns typ

MC961/MC861 = 25 ns typ

OBJECTIVE:

This experiment demonstrates how the Ed-Lab Trainer may be used to demonstrate a wide variety of electrical and electronic principles and circuits.

DESCRIPTION & PROCEDURES:METER

The Meter is a basic 50 microampere meter with an internal resistance of 1,000 ohms. When 50 microamperes is applied through the meter leads marked +I and - , the meter pointer will deflect to full maximum mark on the meter face. The meter terminal marked +E has a 100,000 ohm resistor in series with the +I input. Thus, 5 volts is required across the 100,000 ohm resistor to result in 50 microamperes and the full scale reading corresponds to 5 volts to inputs connected across +E and - . The switch output when the switch is UP is approximately 5 volts. The +P terminal provides the +V voltage on the diagrams.

Connect the potentiometer and meter as shown in Fig. 76-1 and vary the potentiometer and monitor the voltage readings from zero to full scale. Connect the meter as shown in Fig. 76-2 directly. The 1,000 ohm resistance results in a full scale reading of 50 milliamperes. Vary the potentiometer and observe the increased sensitivity. Wire the meter as shown in Fig. 76-3 with the 10,000 ohm resistance (actually the resistance is 10,000 + the meter's 1,000 ohms or 11,000 ohms, but this is ignored at the present time). The meter full scale is now 0.5 volts. Vary the potentiometer and observe the meter movement.

OHM'S LAW

The relationship between voltage, current, and resistance is $V = I R$. Resistors in series are added; $R_s = R_1 + R_2$. Resistors in parallel are calculated in the following manner; $R_p = R_1 R_2 / R_1 + R_2$. Use the circuit in Fig. 76-4 and the meter with the +E (5 volt scale) . Measure the voltage across each resistance and measure the voltage from the power supply. Enter this data in table A. (V_p , V_1 , V_2 , V_3). Calculate the equivalent parallel resistance (3. K and 1K). Calculate the Currents in all the resistors from the actual voltage measurements. Calculate the theoretical voltages V_{1t} , V_{2t} , V_{3t} , from the resistance values. Calculate the theoretical currents I_{1t} , I_{2t} , I_{3t} and I_{4t} . Compare the measured values and the theoretical values. Note that there is an error in the measured values because the meter has a 100,000 ohm resistance which is actually changing the circuit. Using the theoretical values, note that the voltage summed in a loop is equal to zero in accordance with Kirchhoff's Voltage Law and that the sum of all currents to a single point is zero in accordance with Kirchhoff's Current Law.

POTENTIOMETRIC VOLTMETER

Connect the circuit shown in Fig. 76-5 Measure V_p and V_1 . Note that for the equal resistors, V_1 should equal half of V_p . It actually reads less because of the loading of the meter. Connect the circuit shown in Fig. 76-6 The momentary switch is

normally in the UP position and the meter reads the output of the potentiometer setting. When the Momentary Switch M is depressed, the meter reads the voltage difference between the circuit (the two 15 K resistors) and the potentiometer setting. Adjust the potentiometer, while depressing the M switch, until the meter reads zero. At this point, the potentiometer is set at the same voltage as the divider circuit and there is no loading because there is no current in the meter. Release switch M and read the correct voltage on the meter. See if this voltage now reads half of V_p . Readings of this type, by comparing a set voltage with the actual voltage and nulling a meter to effect no loading on the circuit, are called Potentiometric voltmeter techniques.

FLIP - FLOP

The Flip - Flop memory element is shown in Fig. 76- 7. Connect the circuit as shown. Using a loose lead with one end connected to the "0" or ground, touch either output of the Flip-Flop to Set and Clear the memory.

MONOSTABLE MULTIVIBRATOR

The Monostable multivibrator or One-Shot generates a pulse of fixed width based on an input change. The circuit is given in Fig. 76-8. Wire the circuit. The output pulse is started by momentarily touching point A to ground.

POSITIVE TRIGGER

The same monostable from the previous example is triggered from a positive going pulse by momentarily touching point B to "1".

EXCLUSIVE - NOR

The Exclusive-Nor Circuit is shown in Fig. 76-9. The output is the inverse of the Exclusive - Or. The output of the logic is a 1 only when both inputs are the same. Wire the circuit and prepare the truth table.

VOLTAGE CLAMP

A diode acts as a clamping circuit when connected to a reference voltage and limits the output voltage of the logic. Fig. 76-10 shows the connection of a clamping diode to an Inverter circuit. Measure the output voltages with and without the diode.

DIODE CHARACTERISTICS

The diode circuit shown in Fig. 76-11 is forward biased with applied voltage varied by the potentiometer. The 1K series resistor is the other potentiometer. The diode voltage is measured directly across the diode. The diode current is measured by measuring the voltage drop across the series 1K resistor. The current is measured using the +E and - meter inputs resulting in a full scale reading of current of 5 milliamperes. A 10K resistor is placed in series with the +I input resulting in a full scale reading of 0.5 volts for measuring the diode drop using these inputs. The scale may be doubled by varying the series 1K element to 500 ohms (half rotation on the potentiometer) resulting in a full scale reading of 10 milliamperes. Vary the input potentiometer X through all ten marked positions and measure the diode current and voltage and plot the diode characteristics. Use Table B and Graph B.

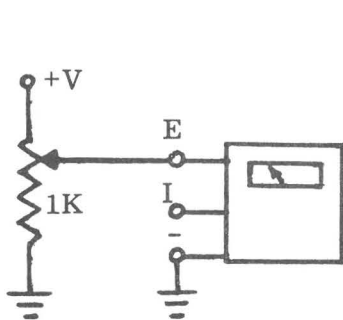


Fig. 76 - 1
5 VOLT METER

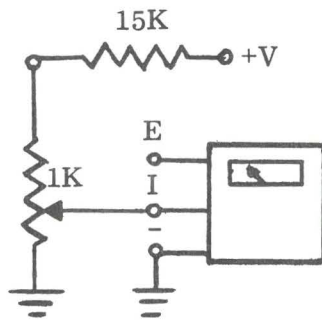


Fig 76 - 2
50 MICROAMP. METER

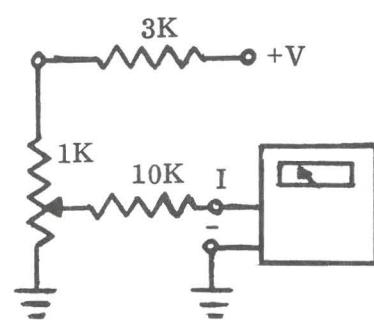


Fig. 76 - 3
0.5 VOLT METER

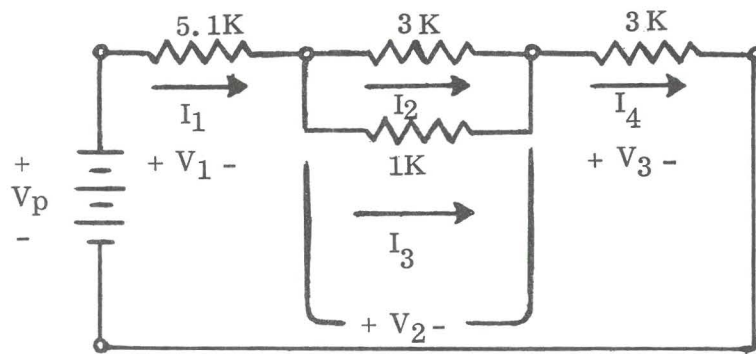


Fig. 4 ELECTRICAL CIRCUIT

TABLE A

	Theoretical value	Actual value
V_p		
V_1		
V_2		
V_3		

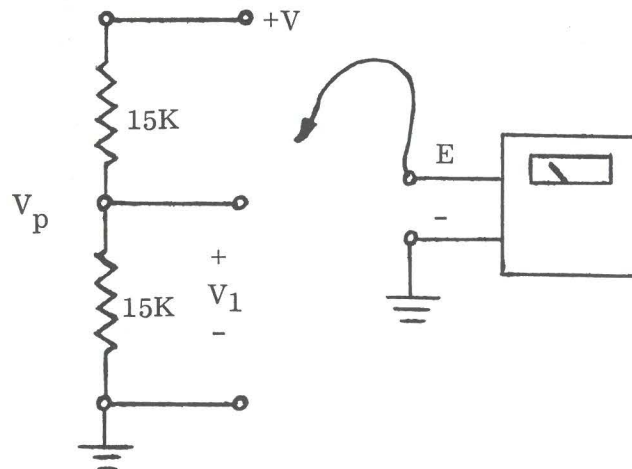


Fig. 75 - 5 VOLTAGE DIVIDER

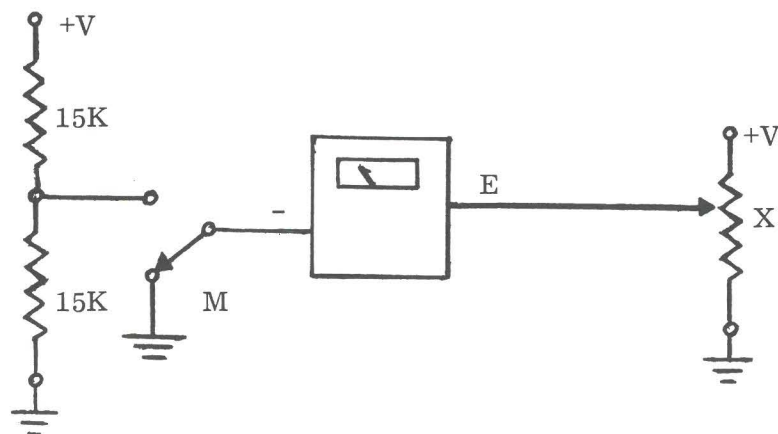


Fig. 75 - 6 POTENTIOMETRIC VOLTMETER

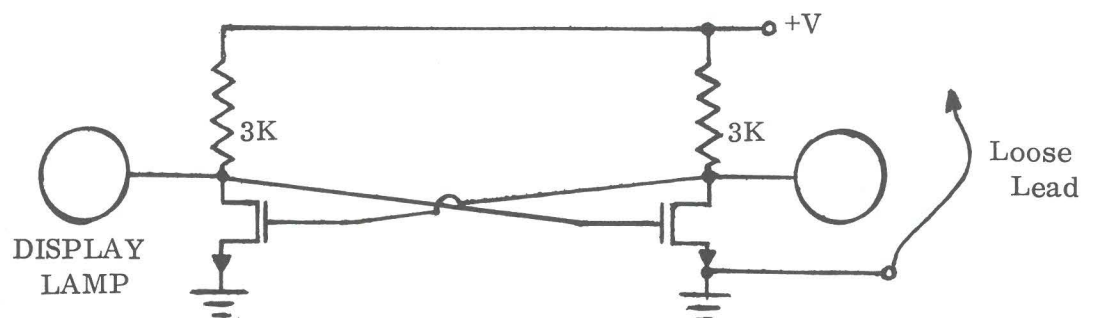


Fig. 76 - 7 FLIP - FLOP

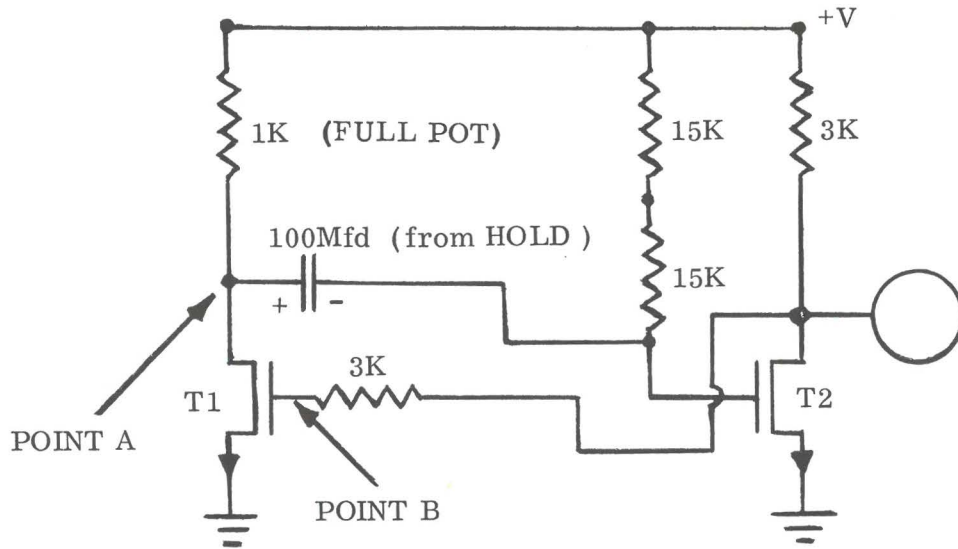


Fig. 76 - 8 MONOSTABLE
MULTIVIBRATOR

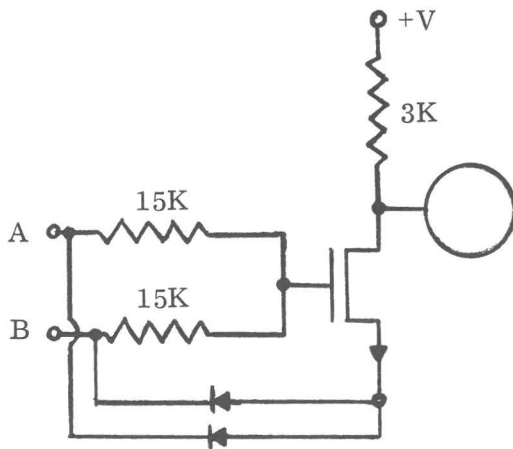


Fig. 76 - 9 EXCLUSIVE - NOR

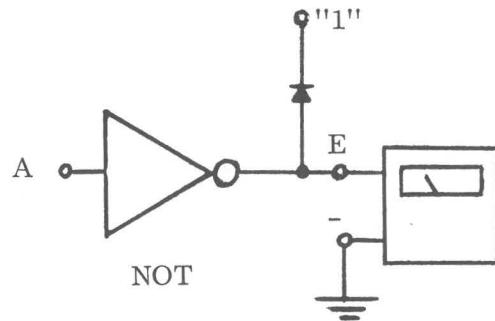


Fig. 76.- 10 VOLTAGE CLAMP

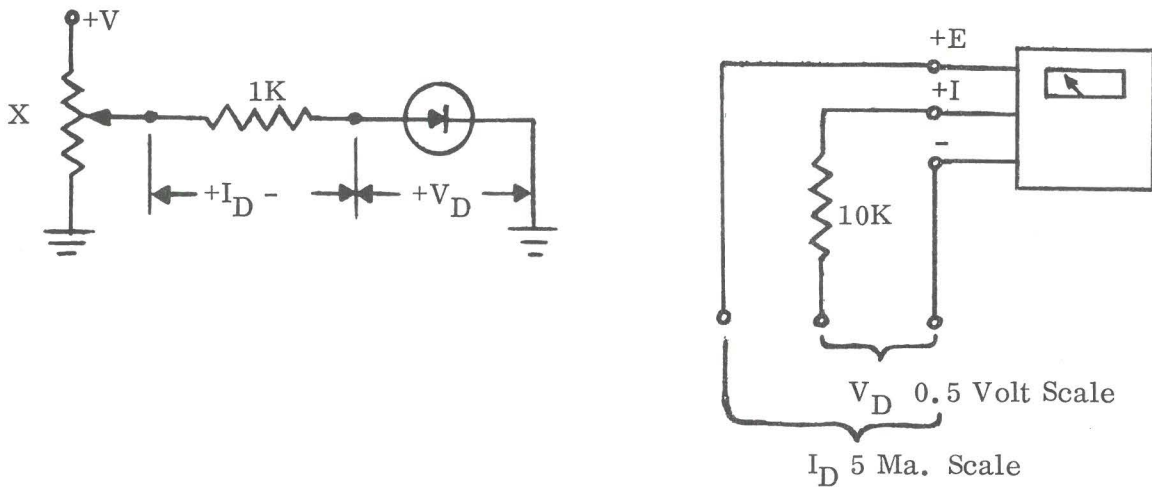
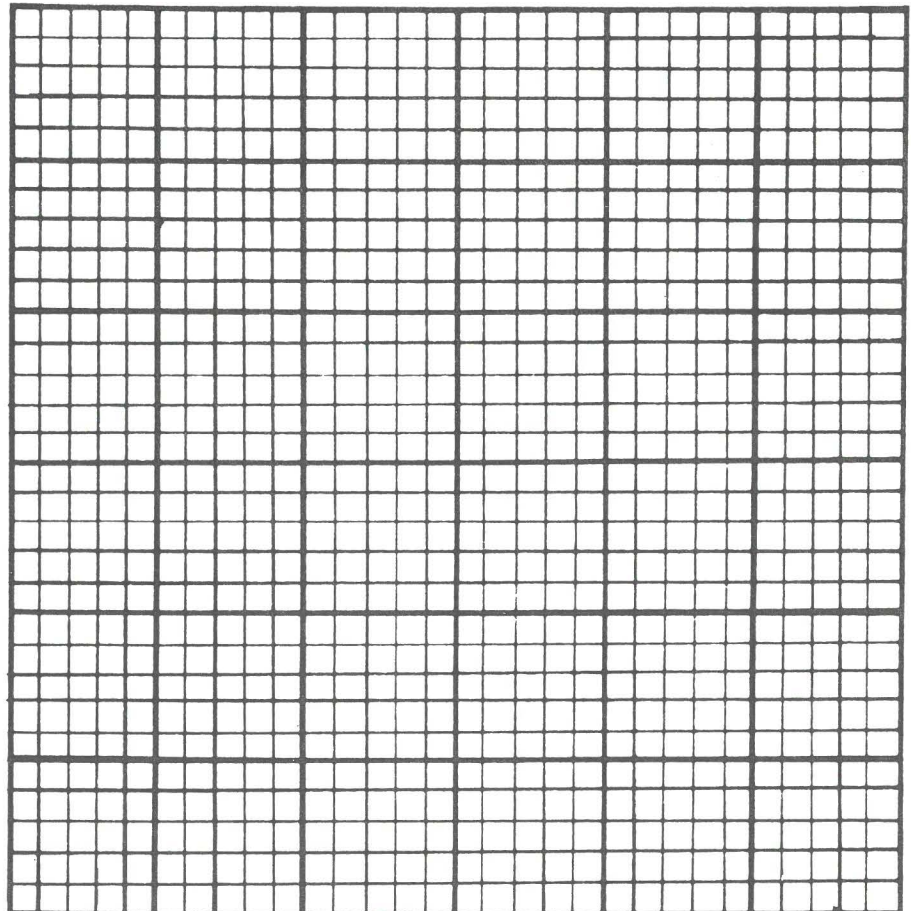


Fig. 76 - 11 DIODE CHARACTERISTICS

TABLE B

[illegible]

GRAPH B



— NOTES —



CES INDUSTRIES, INC.